VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Vzorkování síťových dat

Sampling Network Data

2017 Bc. Dávid Ivan

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Zadání diplomové práce

Student: **Bc. Dávid Ivan**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2612T025 Informatika a výpočetní technika

Téma: Vzorkování síťových dat
Sampling Network Data

Jazyk vypracování: čeština

Zásady pro vypracování:

Analýza tzv. velkých dat je současným fenoménem, kterým se zabývají nejenom vědci, ale také firmy, které mohou díky analýze firemních dat či např. analýzou dat reprezentující chování zákazníků plánovat své budoucí strategie. Pokud je dat velké množství, můžeme místo celé datové kolekce pracovat jen s jejím reprezentativním vzorkem. Cílem práce je implementace jednoho nebo více algoritmů pro tzv. vzorkování (sampling) dat představujících reálné datové kolekce (jako je Web, Internet, sociální sítě, ...). Vzorkování bude použito jako prostředek pro určení globálních vlastnosti takovýchto datových kolekcí.
1. Seznamte se s problematikou komplexních sítí.
2. Seznamte se základními přístupy k vzorkování (samplingu) datových kolekcí reprezentujících reálné sítě.
3. Seznamte se s nejčastěji používanými algoritmy samplingu a proveďte jejich srovnání.
4. Vyberte vhodné algoritmy a naimplementujte je.
5. Navrhněte experimenty a nad zvolenými datovými kolekcemi je proveďte. Experimenty vyhodnoťte.

Seznam doporučené odborné literatury:

[1] M. E. J. Newman, Networks: An Introduction, Oxford University Press (2010), ISBN-10: 0199206651.
[2] Pili Hu, Wing Cheong Lau, A Survey and Taxonomy of Graph Sampling http://arxiv.org/abs/1308.5865
[3] J. Leskovec, Ch. Faloutsos, 2006. Sampling from Large Graphs In proc. of the KDD '06, pp. 631636,
[4] Podle pokynů vedoucího diplomové práce.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **RNDr. Eliška Ochodková, Ph.D.**

Datum zadání: 01.09.2016
Datum odevzdání: 28.04.2017

doc. Dr. Ing. Eduard Sojka
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 28. dubna 2017 .....

Rád bych poděkoval vedoucí mé diplomové práce RNDr. Elišce Ochodkové, Ph.D. za odborné vedení a cenné rady, které mi pomohly tuto práci zkompletovat.

Abstrakt

Diplomová práce popisuje vzorkování grafů a metody, které jsou schopny vytvářet vzorky z reálných sítí. Hlavním cílem práce jsou experimenty nad různými grafovými datovými sadami, jenž jsou vzorkovány popsanými metodami a vyhodnocení výsledků experimentů. Vzorkování grafů je technika výběru podgrafu z původního grafu. V některých případech je dostupný celý graf a cílem je získat vzorek pomocí metod založených na výběru vrcholů nebo hran. V některých případech není celý graf dostupný a vzorkování se provádí s metodami založenými na procházení grafu. Existuje mnoho algoritmů pro výpočet různých metrik grafu, které jsou v případě rozsáhlých grafů výpočetně náročné. V takovém případě je vhodné vytvořit vzorek původního grafu a spustit algoritmus nad tímto vzorkem. Vzorek musí mít obdobné vlastnosti jako původní graf. Sledované vlastnosti grafu a postup měření úspěšnosti metod na základě porovnání vlastností původního grafu a vzorku jsou rovněž v této práci popsány.

Klíčová slova: vzorkování grafu, reálné sítě, rozsáhlé sítě, vzorkovací metody, náhodný výběr vrcholů, procházení grafu, náhodná procházka

Abstract

This master's thesis describes sampling network data and methods, which are able to sample real networks. The main goals of this thesis are experiments on various network data sets, which are sampled by the described methods and evaluation of experimental results. Graph Sampling is technique to pick a subset of vertices or edges from original graph. In some cases, the whole graph is known and the goal of sampling is to obtain a smaller sample by the methods based on vertex or edge selection. In other cases, the graph is unknown and the sample is obtained by the methods based on graph exploration. There are many algorithms to compute various measures of graphs, which are computationally expensive. This is the reason to create sample from large original graph and run a computationally expensive algorithm on this sample. The sample must preserve the properties of the original graph. The observed properties of the graph and process of measuring the success of the sampling methods based on the comparison of the properties are also described in this thesis.

Key Words: graph sampling, real networks, complex networks, sampling methods, random selection of vertices, graph exploration, random walk

Obsah

Seznam použitých zkratek a symbolů **8**

Seznam obrázků **9**

Seznam tabulek **10**

Seznam výpisů zdrojového kódu **11**

1 Úvod **12**

2 Reálné sítě **13**
2.1 Základní pojmy teorie grafů . . . 13
2.2 Vlastnosti reálných sítí . . . 14
2.3 Modely sítí . . . 18

3 Vzorkování **21**
3.1 Vzorkování grafů . . . 21
3.2 Využití vzorkování grafů . . . 21
3.3 Dělení vzorkovacích metod . . . 22
3.4 Vzorkování vzhledem k dostupnosti sítě . . . 23
3.5 Metody pro vzorkování rozsáhlých sítí . . . 24

4 Implementace **34**
4.1 Popis funkčnosti . . . 35
4.2 Diagram tříd . . . 36
4.3 Dodatečné výpočty v jazyce R . . . 38

5 Experimenty **39**
5.1 Zkoumané vlastnosti sítě . . . 39
5.2 Evaluační technika . . . 43
5.3 Popis datových sad . . . 44
5.4 Identifikace nejlepších vzorkovacích metod . . . 46
5.5 Identifikace optimální velikosti vzorku . . . 58
5.6 Experimenty s parametry vzorkovacích metod . . . 59
5.7 Topologická struktura sítě . . . 61

6 Závěr **63**

Literatura **64**

Přílohy **67**

A Grafické znázornění vzorků **68**

B Příloha na CD/DVD **71**

Seznam použitých zkratek a symbolů

API	–	Application Programming Interface (Aplikační Programové Rozhraní)
DBLP	–	DataBase systems and Logic Programming
DS	–	Divided Stratums
FF	–	Forest Fire
FS	–	Frontier Sampling (Multi-dimenzionální Random Walk)
GUI	–	Graphical User Interface (Grafické uživatelské rozhraní)
HEP-PH	–	High Energy Physics - Phenomenology
HYB	–	Hybridní algoritmus
MHRW	–	Metropolis Hastings Random Walk
P2P	–	Peer-to-peer
RDN	–	Random Degree Node
RE	–	Random Edge
RJ	–	Random Jump
RN	–	Random Node
RNE	–	Random Node-Edge
RPN	–	Random PageRank Node
RW	–	Random Walk
SVD	–	singular value decomposition (singulární rozklad matice)
WPF	–	Windows Presentation Foundation
XML	–	eXtensible Markup Language (rozšiřitelný značkovací jazyk)

Seznam obrázků

1 Distribuce stupňů bezškálového a náhodného grafu 15
2 Shluky v sítí vzájemně propojené několika slabými vazbami. 18
3 Vzorkování podle dostupnosti grafu 24
4 Znázornění streamu grafu [21]. 24
5 Proces výběru vrcholů metody Topologically Divided Stratums. 32
6 Ukázka prostředí pro testování vzorků - hlavní okno 34
7 Ukázka prostředí pro testování vzorků - porovnání distribuce 35
8 Diagram tříd základní části a GUI nadstavby. 37
9 Porovnání distribucí stupňů vrcholů 47
10 Distribuce vlastností citační sítě 48
11 Singulární rozklad matice sousednosti - **síť spolupráce (3 lambda)**. 51
12 Distribuce komunit podle Infomap - **síť DBLP** 52
13 Distribuce centralit - **Enron síť**. 53
14 Distribuce stupňů jednotlivých grafů a jejich vzorků (15%) 55
15 Průměrná D-hodnota v závislosti na velikosti vzorku pro bezškálový graf 58
16 Průměrná D-hodnota pro různé velikosti vzorků - **elektrická síť** a **DBLP síť** . 59
17 Průměrná D-hodnota v závislosti na hodnotě parametru. 60
18 Distribuce stupňů - DS metoda s různou hodnotou parametru k 61
19 Vizualizace Enron sítě a vzorků - metody RN, RDN, RE a HYB 68
20 Vizualizace Enron sítě a vzorků - metody RW, RJ, FS, MHRW 69
21 Vizualizace Enron sítě a vzorků - metody DS a FF 70

Seznam tabulek

1 Průměrné D-hodnoty - citační síť 46
2 Průměrné D-hodnoty - bezškálová síť (BA model) 49
3 Průměrné D-hodnoty - náhodná síť 50
4 Průměrné D-hodnoty - síť spolupráce (3-lambda) 50
5 Průměrné D-hodnoty - spoluautorská síť DBLP 52
6 Průměrné D-hodnoty - Enron síť 53
7 Průměrné D-hodnoty - technologická (elektrická) síť 54
8 Globální vlastnosti původních sítí a vzorků - 1. část 56
9 Globální vlasnosti původních sítí a vzorků - 2. část 57
10 Asortativita a modularita u vybraných vzorků. 62

Seznam výpisů zdrojového kódu

1 Random node 25
2 Hybridní algoritmus 26
3 Random Walk 28
4 Forest Fire 29
5 Metropolis-Hastings Random Walk 31
6 Divided Stratums 32
7 Frontier Sampling 33

1 Úvod

Při analýze rozsáhlé sítě hraje velkou roli její velikost. Příliš rozsáhlá síť brání jejímu lepšímu pochopení. Problém nastává již při samotném získání sítě. Kupříkladu služby sociálních sítí nesdílejí kompletní sítě a i když poskytují API pro procházení své sítě, kompletní průchod je obvykle z důvodu omezení nebo rozsáhlé velikosti nemožný. Jen sociální síť Facebook měla v lednu 2017 1,871 miliard uživatelů. Další sociální síť Twitter měla 317 milionů uživatelů [1]. Aktivita těchto uživatelů generuje obrovské sítě, které jsou zajímavé z hlediska analýzy chování uživatelů, identifikování sociálních interakcí nebo zkoumání šíření informací. Výpočet metrik, které síť charakterizují a které se zkoumají při analýze rozsáhlých sítí, mnohdy vyžaduje výpočetně náročně algoritmy. Výpočty nebo simulace nad sítí, která má velký počet vrcholů, jsou výpočetně i časově velmi náročné. Může to být například simulace směrovacích protokolů v internetových sítích, simulace P2P gossip protokolů, simulace propagace virů nebo analýza dopadů virálního marketingu [2]. Vzorkování sítí poskytuje jednoduché, ale účinné řešení, při kterém je z původní sítě vybrán reprezentativní podgraf, který zachovává vlastnosti původní sítě. Při analýze se následně pracuje se vzorkem mnohem menší velikosti, který zastupuje původní síť.

Tato práce je zaměřená na vzorkování sítí, jejichž účelem je získání reprezentativních vzorků, které mohou být použity namísto původní sítě, například při výpočetně náročných operacích nad grafem. Cílem práce je přehledně popsat a implementovat některé dostupné vzorkovací metody a určit nejlepší metody, které vytvářejí vzorky z vybraných síťových datových sad. Úspěšnost metody je určena tím, jak dokáže vytvářet vzorky s co možná nejpodobnějšími vlastnostmi původní sítě.

První část práce je zaměřená na obecný popis reálných sítí a jejich vlastností. Rovněž jsou popsány i konkrétní distribuce vlastností, které slouží pro určení míry podobnosti dvou grafů. Vzorky musí co nejlépe zachovávat původní vlastnosti původní sítě. Druhá část se zabývá samotným vzorkováním - důvody a kde lze využít vzorkování. Popisuje rovněž jednotlivé metody pro generování vzorků, seskupené podle dostupnosti původní sítě. Třetí část popisuje implementační část práce, výsledkem které je nástroj pro vzorkování sítí a určování podobností jednotlivých vlastností dvou grafů.

Poslední část je experimentální a popisuje několik sítí, nad kterými byly provedeny experimenty s různými vzorkovacími metodami. Obsahuje popis evaluace jednotlivých vzorků a výsledky experimentů. Hlavním cílem je vytvoření vzorků z vybraných sítí pomocí implementovaných metod a vyhodnocení kvality vzorků. Experimentální část se dál zabývá optimální velikostí vzorku a chování vzorkovacích metod s ohledem na topologickou strukturu sítí. Součástí experimentální části jsou globální vlastnosti původních sítí i vzorků a série grafů a tabulek, které porovnávají různé distribuce vlastnosti grafů jak statisticky, tak i vizuálně.

2 Reálné sítě

Existuje řada opakujících se vzorů, vyskytujících se v síťových strukturách, které mají zásadní vliv na to, jak tyto sítě pracují. I když reálné sítě mohou pocházet z různých oblastí, jako například sociální sítě, informační sítě či biologické sítě, vykazují určité společné rysy.

Pod pojmem reálná síť si lze konkrétně představit například architekturu počítačové sítě, která se skládá z uzlů typu počítač, server nebo síťový prvek, jenž jsou spolu vzájemně propojené. Dalším příkladem reálné sítě může být sociální síť Facebook, ve které jsou dva uživatelé propojení, jestliže spolu kamarádí. Zástupcem biologické reálné sítě může být proteinová interakcí síť, která zachycuje interakci proteinů v buňkách [1].

Reálná síť má mnoho vlastností, jejichž analýzou lze charakterizovat její strukturu. Jedná se zejména o velikost komponent, délky cest mezi vrcholy, distribuce stupňů vrcholů a shlukovací koeficient. Tyto vlastnosti reálných sítí budou popsány v následujících odstavcích.

Pojem **komplexní síť** označuje rozsáhlé síťové struktury, jejichž entity jsou výrazně propojené. Ať už se jedná o internet, sociální sítě, biologické sítě nebo citační sítě, jejich charakteristikou je komplexnost. Struktura komplexní sítě je nepravidelná, komplexní a dynamicky se vyvíjející v čase. Je obtížné odvodit kolektivní chování celé sítě ze znalostí jednotlivých komponent sítě. Protože se komplexní systémy stávají důležitými ve vědě, ekonomice i v každodenním životě, je jejich pochopení, matematický popis a predikce vývoje jedním z hlavních intelektuálních a vědeckých výzev 21.století [3].

2.1 Základní pojmy teorie grafů

Jelikož se na jednotlivé entity lze dívat jako na vrcholy grafu a vztahy mezi entitami se dají vyjádřit vazbou, která představuje hranu mezi vrcholy, komplexní struktury je možné modelovat prostřednictvím grafů.

Neorientovaný graf je definován jako uspořádaná dvojice $G = (V, E)$, kde V je množina vrcholů a E je množina hran (neuspořádaných dvojic $u, v \in V$) a platí, že $E \subseteq P_2(V)$ - množina vybraných dvouprvkových podmnožin množiny vrcholů V.

Vrcholy $u, v \in V$ se nazývají **sousední**, jestliže mezi nimi existuje hrana $\{u, v\} \in E$, která je spojuje. Vrcholy u, v jsou tak **incidentní** s hranou $\{u, v\}$. **Stupeň vrcholu** v je počet hran incidentních s daným vrcholem a značí se k_v.

Orientovaný graf je opět definován jako uspořádaná dvojice $G = (V, E)$, ale E je množina orientovaných hran - množina uspořádaných dvojic $u, v \in V$. U orientovaného grafu se rozlišuje vstupní a výstupní hrana a vstupní a výstupní stupeň vrcholu. Vstupní stupeň je definován jako počet vstupních hran vrcholu a výstupní stupeň jako počet výstupních hran vrcholu.

[1] `http://string-db.org/`

Podgraf grafu $G = (V, E)$ vznikne vymazáním některých (nebo také žádných) vrcholů původního grafu a všech hran do těchto vrcholů zasahujících. Mohou být vymazány i jen některé hrany. Značí se $G' = (V', E')$, kde $V' \subseteq V$ a $E' \subseteq E$ [4].

Další definice potřebné v této práci budou podle potřeby uvedené dále v textu.

2.2 Vlastnosti reálných sítí

2.2.1 Centra

V reálných sítích se mohou vrcholy dělit do dvou skupin: vrcholy, které nesou užitečnou informaci se nazývají autority a centra na tyto autority odkazují a určují, kde je lze najít [5]. Příkladem může být citační síť. Vědecký článek obsahuje shrnutí jiných vědeckých prací, na které se pomocí citací odkazuje. Odkazované práce obsahují podrobnější informace o daném tématu.

Malcolm Gladwell ve své knize [6] provedl test, ve kterém náhodně vybral 248 příjmení z telefonního seznamu a za každé, které testovaná osoba znala, dostala bod. Otestoval několik rozličných sociálních skupin a zjistil, že rozpětí výsledků bylo překvapivě velké. Například pro skupinu vysokoškoláků bylo rozpětí od 9 do 118. Osoby s vysokým počtem bodů nazval prostředníky. Prostřednicí mají výjimečnou schopnost získávání přátel a známých. Plní důležitou roli, protože spojují různě odlišné skupiny, vytvářejí trendy a módní vlny [6].

V sítí vědecké spolupráce je Pál Erdős považován za jednoho z významných center sítě, protože během svého života publikoval kolem 1500 článků s 511 spoluautory. Erdősovo číslo říká, jakou vzdálenost má autor v sítí vědecké spolupráce od Erdőse. Více než 90% publikujících vědců mělo Erdősovo číslo menší nebo rovné 8 bez ohledu na obor, ve kterém pracovali. Průměr pro všechny je 4,65 [7].

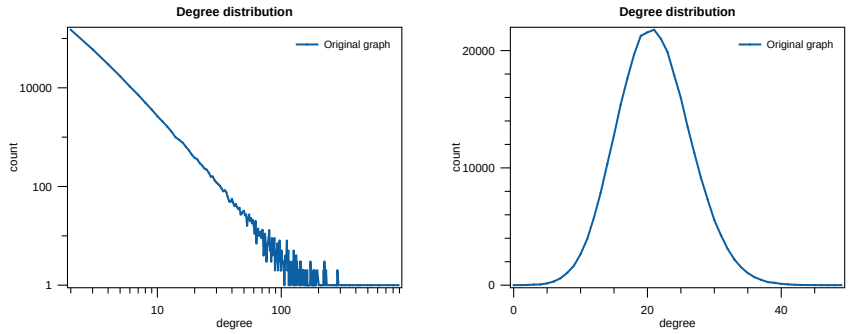
Centra se vyskytují v reálných sítích a jsou důležitým prvkem, protože formují strukturu sítě. Představují mosty mezi dvěma místy sítě, a tak je jakýkoliv vrchol vzdálen od centra často jen na jeden nebo dva kroky.

2.2.2 Mocninný zákon

Jedním ze základních vlastností sítě je distribuce stupňů vrcholů. Jak již bylo popsáno v kapitole 2.1, stupeň vrcholu v k_v značí počet hran incidentních s daným vrcholem v. **Distribuce stupňů** je funkce $P(k)$, která udává pravděpodobnost, že náhodně vybraný vrchol má stupeň k. U náhodného grafu (viz 2.3.2) je distribuce stupňů binomická a většina vrcholů má stejný počet incidentních hran. Pro velké množství vrcholů se binomické rozdělení aproximuje Poissonovým rozdělením. To se řídí Gaussovou zvonovitou křivkou, která má výrazné maximum a všechny vrcholy mají podobný průměrný stupeň. Odchylky od průměru jsou vzácné (viz obr 1 b). Důsledkem je, že u náhodných grafů jsou vrcholy stejně propojené a neexistují vrcholy s velkým stupněm, které se označují jako centra (viz 2.2.1).

Maďarský vědec Albert-László Barabási uvádí, že v řadě reálných sítí jsou to právě mocninné zákony, které umožňují vznik většího množství center [8]. Barabási sesbíral informace o 325 000

stránkách na doméně nd.edu a zjistil, že v síti stránek se rozdělení distribuce stupňů vůbec neřídí Gaussovou křivkou. Naopak, na 82% stránek odkazovalo 3 a méně jiných stránek a na 24 jiných stránek mířilo více než 1000 jiných stránek [8]. Toto zjištění do jisté míry koreluje s Paretovým pravidlem 80/20, podle kterého 80% důsledků pramení z 20% příčin - například 80% bohatství vlastní 20% lidí [9]. Toto pravidlo, stejně tak i distribuce stupňů v reálných sítích, se řídí mocninným zákonem. Mocninný zákon umožňuje vznik center s velmi velkým stupněm, protože na rozdíl od Gaussovy křivky, mocninná křivka klesá mnohem pomaleji.



(a) Mocninný zákon (log měřítko)

(b) Poissonovo rozdělení (standartní měřítko)

Obrázek 1: Znázornění distribuce stupňů bezškálového grafu a náhodného grafu.

Sítě respektující mocninný zákon se označují jako bezškálové, protože v nich neexistuje typická hodnota stupně uzlu (škála), tak jako tomu je u náhodných sítí s Poissonovým rozdělením. Rozdíl mezi mocninným zákonem s „dlouhým ocasem“ a Poissonovým rozdělením je ukázán na obrázku 1. K tomu, aby se síť chovala podle mocninného zákona, stačí aby rostla a nové uzly se nepřipojovaly náhodně, ale preferenčně - k těm, které už mají více vazeb. Princip „bohatí bohatnou a chudí chudnou“ umožňuje, aby v síti vznikala centra.

Mocninná funkce je definována vztahem 1, kde funkce $P(k)$ určuje, kolik vrcholů se stupněm k existuje a γ je exponent konektivity.

$$P(k) = k^{-\gamma} \tag{1}$$

Exponent konektivity říká, kolikrát méně je v síti uzlů s velkým stupněm, ve srovnání s vrcholy s nízkým stupněm. Zlogaritmováním funkce 1 vznikne funkce 2, kde $log(P(k))$ je lineárně závislé na $log(k)$.

$$log(P(k)) = -\gamma \cdot log(k) \tag{2}$$

Sklon této lineární přímky je dán exponentem γ. V logaritmickém měřítku má mocninná křivka tvar přímky [3].

Bylo zjištěno, že bezškálové sítě mají hodnotu exponentu konektivity γ v rozmezí $2 < \gamma < 3$. Při experimentu provedeném na vzorku stránek z domény nd.edu bylo zjištěno, že exponent u distribuce vstupních stupňů vrcholů byl roven $\gamma_{in} = 2,1$ a exponent u distribuce výstupních stupňů zase $\gamma_{out} = 2,45$ [8].

2.2.3 Fenomén malého světa

Small World efekt je hypotéza, že průměrná vzdálenost l v reálných sítích je malá a platí vzorec 3, kde N je počet vrcholů sítě.

$$l \propto lnN \tag{3}$$

U bezškálových sítí s exponentem konektivity $2 < \gamma < 3$ se zjistilo, že průměrná vzdálenost l s rostoucím počtem vrcholů stoupá ještě pomaleji a platí vzorec 4.

$$l \propto lnlnN \tag{4}$$

Je to dáno tím, že v bezškálových sítích existují tzv. huby, které radikálně zkracují vzdálenosti nejkratších cest. Tato vlastnost se označuje jako Ultra-Small World. Pro příklad, bezškálová síť o velikosti $7x10^9$ vrcholů má průměrnou vzdálenost pouze 3,12 [3].

Vzdálenost v grafech (grafová metrika) je vzdálenost dvou vrcholů u a v a je to délka nejkratší cesty mezi u a v, značí se jako $d(u, v)$. Číslo $c(e)$ se nazývá délka (ohodnocení) hrany e. Délku nejkratší cesty definuje vzorec 5, kde k je počet hran mezi vrcholy. Není-li hrana e ohodnocená, je délka hrany rovna jedné [4].

$$\sum_{i=1}^{k} c(e_i) \tag{5}$$

Průměrná vzdálenost je průměrná délka nejkratších cest mezi všemi dvojicemi vrcholů sítě a počítá se pomocí vzorce 6. Neexistuje-li cesta mezi vrcholy u a v nebo $u = v$, pak je vzdálenost nulová.

$$l = \frac{1}{N(N-1)} \sum_{u,v \in V} d(u, v) \tag{6}$$

Pravděpodobně první pokus, který vedl k zjištění, že reálné sítě mají malou průměrnou vzdálenost, provedl Stanley Milgram z Harvardovy univerzity v roce 1967. Napsal větší množství dopisů adresovaných svému příteli v Bostonu a rozdal je náhodně vybraným lidem v Nebrasce. Každý byl požádán, aby se pokusil doručit dopis jen prostřednictvím osob, které osobně zná. Ze 160 dopisů jich bylo doručeno 44. Počet prostředníků se pohyboval od 2 do 10 a průměrný počet prostředníků byl **6** [10]. Z toho Milgram odvodil, že průměrná vzdálenost mezi dvěma náhodně vybranými lidmi je pouze šest. Jeho slogan „šest stupňů odloučení“ si od té doby získal určitou

popularitu ve všeobecném podvědomí [11]. Z práce [10] také pochází pojmenování tohoto jevu - Small World.

V sociální sítí Facebook byla v únoru 2016 mezi všemi 1,59 miliardy aktivními uživateli stanovena průměrná vzdálenost na 3,57. Ještě v roce 2011 se 721 miliony aktivními uživateli byla průměrná vzdálenost 3,74. Jak síť rostla, uživatelé se stávali více propojenými a vzdálenost se zkrátila [12]. Barabási na vzorku 350 000 stránek zjistil, že průměrný počet kroků (délka nejdelší orientované cesty) mezi stránkami je roven 11. Při zkoumání většího počtu dokumentů dospěl k závěru, že průměrný počet kroků roste mnohem pomaleji, než počet dokumentů a řídí se jednoduchým pravidlem (vzorec 3). Průměrná vzdálenost stránek v celém webu v roce 1999 tak byla odhadnuta na 19 [8]. I další experimenty ukázaly, že reálné sítě mají malou průměrnou vzdálenost vzhledem k jejich velikosti. V citačních sítích jsou od sebe osoby vzdálené 4 až 6 vazbami. Průměrná vzdálenost v síti článků na Wikipedii je 3,45. Small World efekt je proto typickou vlastností reálných sítí.

2.2.4 Shluky v sítích

Na rozdíl od náhodné sítě, v reálných sítích existují komunity - shluky vrcholů, které jsou spolu vzájemně vysoce propojené. Tyto shluky jsou navzájem propojené několika slabými vazbami, podobně jako na obrázku 2. Autorem této teorie silných vazeb je Mark Granovetter. Byl přesvědčen, že naše povrchní a vzdálené známosti hrají často důležitou roli. Mají funkci mostů, které nás propojují s lidmi z úplně jiných oblastí. Granovetter pro dokázání své teorie uskutečnil výzkum. Oslovil osoby, které se při shánění práce spoléhali na doporučení jiného člověka. Položil jim otázku „Jak často jste potkávali tuto osobu předtím, než jste nastoupili do nového zaměstnání?“. 83% dotázaných odpovědělo, že jen občas nebo dokonce zřídka. Většina uchazečů se spoléhala na pomoc od téměř cizích lidí [13].

Hustotu propojení všech sousedů daného vrcholu reprezentuje **shlukovací koeficient**. Čím vyšší má daný vrchol shlukovací koeficient, tím více je součástí komunity. Shlukovací koeficient měří průměrnou pravděpodobnost, že dva sousední vrcholy jiného vrcholu jsou spolu rovněž propojeny. Měří tedy hustotu trojúhelníků v síti. Shlukovací koeficient pro celou síť je definován vzorcem 7. Je možné jej definovat i pro jednotlivé vrcholy pomocí vztahu 8 [5].

$$C = \frac{3 \cdot \text{počet všech trojúhelníků}}{\text{počet trojic propojených dvěma hranami}} \tag{7}$$

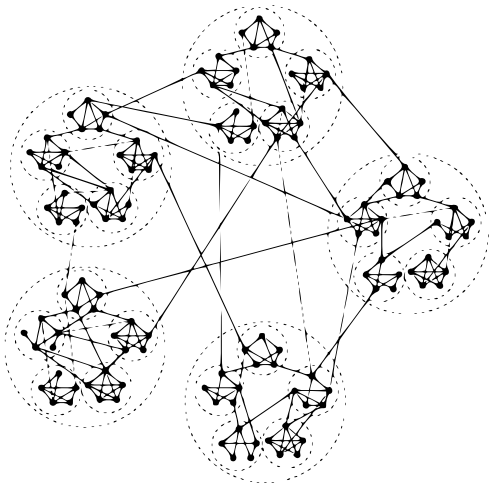
$$C_i = \frac{\text{počet párů sousedů vrcholu i, které jsou propojené}}{\text{počet párů sousedů vrcholu i}} \tag{8}$$

Jiný způsob výpočtu je definován vzorcem 9, kde L_i je počet hran mezi sousedy vrcholu i a k_i je počet sousedů vrcholu i. Shlukovací koeficient celé sítě je průměrný shlukovací koeficient

daný vzorcem 10, kde N je počet vrcholů [3].

$$C_i = \frac{2L_i}{k_i(k_i - 1)} \tag{9}$$

$$C = \frac{1}{N} \sum_{i=1}^{N} C_i \tag{10}$$



Obrázek 2: Shluky v sítí vzájemně propojené několika slabými vazbami.

Mnoho reálných sítí vykazuje vysoký shlukovací koeficient. V knize [5] je k dispozici měření několika sítí z různých zdrojů a většina jich má shlukovací koeficient mezi 0,2 až 0,6. Například síť filmových herců o velikosti 450000 vrcholů má koeficient shlukování $C = 0,2$. Síť spolupráce mezi fyziky o velikosti 53000 vrcholů má $C = 0,45$. Jedná se o sítě s velkým počtem trojúhelníků. Kdyby byla síť spolupráce mezi fyziky propojována náhodně, byl by shlukovací koeficient asi 100x nižší a se zvětšujícím se počtem vrcholů by se shlukovací koeficient zmenšoval [5]. To u reálných sítí neplatí.

2.3 Modely sítí

Tato práce využívá ke generování testovacích sítí různé modely, kterou budou v této kapitole popsány. Pro uvedení do problematiky je také v krátkosti popsána historie teorie grafů.

2.3.1 Historie teorie grafů

Základy teorie sítí položil v první polovině 18. století švýcarský matematik Leonhard Paul Euler, když se zabýval problémem nazvaným „sedm mostů města Královce“. Pět mostů propojovalo ostrov mezi dvěma rameny řeky s ostatními částmi města a další dva mosty překlenovala dvě říční ramena. Problém se zabýval otázkou, zda je možné přejít všech sedm mostů a nejít po žádném z nich dvakrát. Euler v roce 1736 předložil matematický důkaz, že v případě sedmi

mostů žádná taková cesta neexistuje. Město si představil jako graf, ve kterém jednotlivé části města reprezentoval uzlem a mosty hranou [14].

Teorie grafů zažila po Eulerovi prudký rozmach. Až do poloviny 20. století bylo cílem teorie grafů objevovat a katalogizovat vlastnosti různých typů grafů. Byly vyřešeny problémy, jako například hledání únikové cesty z bludiště v roce 1873 nebo hledání posloupnosti kroků jezdce na šachovnici tak, aby každé políčko bylo navštíveno právě jednou, a aby se jezdec vrátil na výchozí pozici.

V 50. letech 20. století dva maďarští matematici Pál Erdős a Alfréd Rényi položili se svým modelem základy teorie náhodných sítí. Jejich práce se poprvé v historii zabývala otázkou vzájemně provázaného světa - jak vůbec sítě vznikají.

2.3.2 Erdős–Rényi model

Model náhodné sítě se objevil jako vůbec první a matematici Erdős a Rényi se s ním pokusili popsat reálné sítě. Množství uzlů n je pevně dané a mezi tyto uzly se s uniformní pravděpodobností vloží m hran. Tato varianta je označována jako $G(n, m)$. Druhá varianta $G(n, p)$ má opět pevný počet uzlů n a p udává pravděpodobnost spojení libovolných dvou uzlů hranou [15].

Jelikož propojení mezi uzly vzniká náhodně, distribuce stupňů vrcholů má Poissonovo rozdělení. Koeficient shlukování je malý a odpovídá pravděpodobnosti p. Rovněž průměrná vzdálenost je malá (počítáno na největší komponentě) a v náhodné síti se nevyskytují centra. Náhodný model tak není vhodný pro modelování reálných sítí.

2.3.3 Watts-Strogatz model

Watts-Strogatz model generuje náhodné sítě s vlastnostmi malého světa. Model Erdőse a Rényiho neuměl generovat sítě s přítomností shluků, které jsou typické pro malý svět. Proto Duncan Watts a Steve Strogatz navrhli v roce 1998 alternativu k modelu náhodné sítě, která generuje síť se shluky s vysokým průměrným koeficientem shlukování a přitom zachovává krátké vzdálenosti [16].

Algoritmus modelu s předpisem $WS(n, k, \beta)$ v prvním kroku vygeneruje n vrcholů uspořádaných do kruhu. Každý vrchol je propojen neorientovanými hranami s k nejbližšími sousedy. V druhém kroku je každá hrana s pravděpodobností β přepojena k jinému, náhodně vybranému vrcholu kromě původních sousedních vrcholů tak, aby nevznikla smyčka, ani dvojitá hrana. Pro velmi nízké hodnoty β má síť vlastnosti malého světa. Pro hodnoty β blížící se k 1, je síť podobná náhodné síti.

Distribuce stupňů opět odpovídá Poissonovu rozdělení. Je zachována malá průměrná vzdálenost a shlukovací koeficient je vysoký. V sítích malého světa se nevyskytují centra. Absenci center řeší další model.

2.3.4 Barabási-Albert model

Jak již bylo popsáno v kapitole 2.2, pro reálné sítě je typický efekt malého světa, shlukování, existence center a mocninné rozdělení distribuce stupňů. Watts-Strogatz model je schopen vysvětlit efekt malého světa a shluky v sítích. Avšak existence center a mocninné rozdělení stupňů vrcholů jsou u tohoto modelu vyloučeny. Až sítě vytvořené pomocí Barabási-Albert modelu mají všechny výše zmíněné vlastnosti reálných sítí. Model je nazván podle dvojice maďarských vědců Albert-László Barabási a Réka Albert, kteří vytvořili algoritmus generující **bezškálové sítě**, pro které platí [3]:

- **Neustálý růst sítě** - síť při zrodu obsahuje jen několik málo uzlů a v každém kroku se rozrůstá přidáváním nových uzlů. Síť je dynamická a neustále se proměňuje. Příkladem může být síť Internetu nebo citační sítě autorů, které se neustále rozrůstají.
- **Preferenční připojování** - při každé iteraci se nové uzly nepřipojují ke stávajícím vrcholům náhodně, ale jsou preferovány vrcholy s větším stupněm. Vrchol s dvakrát větším stupněm má dvakrát větší šanci, že se nový uzel připojí právě k němu. Nové vrcholy jsou připojovány k těch starším a novější uzly jsou znevýhodňovány. To má za následek vznik center a bezškálová síť dodržuje mocninný zákon.

Síť je nazývána bezškálovou, pokud distribuce stupňů odpovídá mocninnému rozdělení s exponentem konektivity $2 < \gamma < 3$. Dále mají bezškálové sítě vysoce propojená centra, která drží sítě pohromadě a činí sítě odolné vůči náhodným chybám. Problém může nastat při cíleném útoku na centra, který zapříčiní rychlý rozpad sítě. Distribuce stupňů u bezškálových sítí zůstává zachována i při náhodném přepojování hran [17].

Postup generování je následující: v prvním kroku se vytvoří počáteční síť s m_0 vrcholy a takovým počtem hran, aby byla počáteční síť souvislá. V každé iteraci se připojuje nový vrchol k m existujícím vrcholům, přičemž musí platit, že $m \leq m_0$. Pravděpodobnost, že se připojí ke stávajícímu uzlu i s k_i stupněm, je dána vztahem 11, kde k_j značí stupeň již existujícího vrcholu j.

$$p_i = \frac{k_i}{\sum\limits_j k_j} \tag{11}$$

Průměrná vzdálenost u bezškálové sítě je nízká. Pro exponent konektivity $2 < \gamma < 3$ je průměrná vzdálenost dána vztahem 4 a pro $\gamma = 3$ vztahem 12.

$$l \propto \frac{ln(N)}{ln(ln(N))} \tag{12}$$

Koeficient shlukování u bezškálových sítí je dán vztahem 13, kde N je počet vrcholů [3].

$$C = \frac{(ln(N))^2}{N} \tag{13}$$

3 Vzorkování

Vzorkování dat obecně je technika statistické analýzy, která slouží k vytvoření reprezentativního vzorku z velkých datových kolekcí. Umožňuje pracovat s menšími daty tam, kde je nemožné efektivně analyzovat úplnou datovou sadu. Typické použití vzorkování je například při dolování dat, zejména z tzv. big dat, které jsou charakteristické často se měnícím velkým objemem strukturovaných i nestrukturovaných dat [18].

Ve statistice slouží vzorkování k výběru podmnožiny jedinců, která je následně použita k odhadování vlastností celé populace. Tato práce se zabývá síťovými daty, proto bude dál v textu podrobně popsáno vzorkování sítí a různé vzorkovací metody pro vzorování sítí.

3.1 Vzorkování grafů

Existuje mnoho globálních či lokálních vlastností sítě, které jsou důležité pro pochopení sítě a které síť charakterizují. Může to být distribuce stupňů vrcholů, průměrná vzdálenost nebo například průměr sítě. Je dobré vědět, zda vzorek, který je odvozen od původního grafu, zachovává tyto vlastnosti původního grafu. Pokud ano, spuštění nějakého algoritmu nad vzorkem může mít stejný efekt s podobným výsledkem, jako spuštění nad původním grafem a to s nižší časovou náročností. Některé algoritmy jsou výpočetně náročné a běh nad velmi velkým grafem může trvat neúměrné dlouhou dobu. Jedná se většinou o NP-úplné úlohy, jako například hledání nejkratší cesty nebo výpočet centralit. Navíc, když je známé, jaké výsledky produkuje daná vzorkovací metoda, lze poměrně přesně vypočíst vlastnosti původního grafu z vypočtených vlastností vzorku.

Vzorkování grafů je jedna z jednoduchých ale efektivních možností transformace grafu. Je založena na výběru podmnožiny vrcholů $V' \subseteq V$ a hran $E' \subseteq E$ z původního grafu $G = (V, E)$. Výsledkem je vzorek $G' = (V', E')$. Výhodou této metody je, že algoritmy pro vzorkování jsou jednoduché a časově efektivní. Vzorkování netrvá déle, než výpočty žádaných parametrů přímo nad původním grafem.

3.2 Využití vzorkování grafů

I když může vzorkování sloužit k různým využitím, všechny mají společné dvě vlastnosti: velikost původního grafu je během vzorkování zmenšena a ze vzorku lze vyčíst vlastnosti podobné původnímu grafu. Následující konkrétní ukázky využití vzorkování tyto vlastnosti splňují [2, 19]:

- **Nedostatečný přístup k datům** - například v sociální síti Facebook nelze za účelem získání sítě procházet kvůli limitům API všechny osoby. Proto se náhodně vybere osoba a od této osoby se přechází po hranách k dalším osobám. Dochází tak k vzorkování vrcholů pomocí metody založené na náhodné procházce. Je důležité dobře odhadnout, kolik vrcholů se musí navštívit tak, aby se ze vzorku získala relevantní informace.

- **Průzkum skryté populace** - používá se v sociologických průzkumech, kdy je nutné získat informace o skryté skupině osob, například skupina drogové závislých lidí. Přímý přístup k těmto skupinám je většinou nemožný. Začíná se s malým počtem osob, od kterých se získávají informace o dalších členech skupiny. Typickou metodou je Snow Ball vzorkování, ve kterém se na vrcholy postupně nabalují další vrcholy podle vazeb mezi předchozími vrcholy.

- **Sparsifikace grafu** - Mnoho sítí je příliš velkých a manipulace s nimi je obtížná. Řešením je aproximovat husté sítě řidšími sítěmi. To zahrnuje jak redukci hran, tak i redukci vrcholů. Často jsou na výsledný graf sparsifikace kladeny přísné požadavky. Například, že všechny vrcholové řezy původního grafu musí být zachovány.

- **Snížení ceny testů** - Síť interakcí proteinů je často studována v biochemických pracích. Přesný test interakcí mezi všemi možnými páry proteinů může být výpočetně náročný. Řešením je testovat páry proteinů jen ve vzorku. Jiný způsob využívá priorizaci spojení, při kterém se jako první testují nejdůležitější spojení. I k nalezení prioritních spojení může být použit vzorek.

- **Vizualizace** - Originální síť může být příliš velká k vizualizaci, nemusí se vejít na obrazovku a zobrazení všech hran vede k nepřehlednosti. Tím se ztrácí informace, která by mohla být vizualizací sítě poskytnuta. Cílem vzorkování je menší graf, který vypadá stejně jako původní graf. Pokud je vzorek „kvalitní“, lze například vykreslit komunity sítě, které odpovídají i původnímu grafu.

- **Nedostatek paměti a výpočtově složité operace** - Analýza sítě může být výpočetně a časově náročná a na kompletní rozsáhlé síti nemusí dojít k výpočtům v rozumném čase. Také přístup k celé síti nemusí být možný. Velikost části World Wide Webu, která je indexovaná Googlem, je odhadována na více než 48 miliard vrcholů [20]. Je nemožné jen uložit tak rozsáhlou síť do paměti, natož nad ní provádět složitější výpočty. Vytvořit vzorek a pracovat s ním je možností, jak nepřímo analyzovat celou síť. V těchto případech je vhodné použití metod, které jsou založené na náhodných procházkách.

3.3 Dělení vzorkovacích metod

Pro správné vytváření vzorků je nutné zodpovědět následující otázky: Jakou použít metodu pro vzorkování a jak velký vzorek vytvořit? Správný výběr záleží také na typu původního grafu. Lze použít metody založené na

1. výběru vrcholů
2. výběru hran
3. prohledávání grafu

Velikost grafu je měřena počtem vrcholů a cílem je zjistit, při jaké velikosti je ještě vzorek podobný původnímu grafu. Jak lze měřit úspěšnost vzorkování a kvalitu výsledného vzorku? V kapitole 5.1 je uvedeno několik distribucí vlastností, které se budou porovnávat, jak empiricky, tak i statisticky. Přesná metoda pro určení podobnosti je uvedena v kapitole 5.2.

Podle [2] může být vzorek využit dvěma způsoby:

1. U **scale-down** vzorkování se porovná vzorek s původním grafem. Rozsáhlý statický neorientovaný graf G s n vrcholy je převzorkován na graf S s n' vrcholy, kde $n' \ll n$. Cílem je to, aby měl vzorek S co nejvíc podobné vlastnosti (s přihlédnutím k měřítku) jako graf G. Může to být například podobná distribuce stupňů nebo průměr grafu. Na tento způsob využití vzorkování je zaměřena praktická část práce.

2. V případě **back-in-time** vzorkování je cílem napodobit stav grafu G v určité časové chvíli jeho vývoje vzorkem S. Je známa jen výsledná statická podoba grafu G a nelze zjistit stáří vrcholů a hran. $G_{n'}$ představuje graf G v době, kdy měl n' vrcholů. Cílem je nalézt graf S s n' vrcholy tak, aby měl podobné vlastnosti jako graf $G_{n'}$.

3.4 Vzorkování vzhledem k dostupnosti sítě

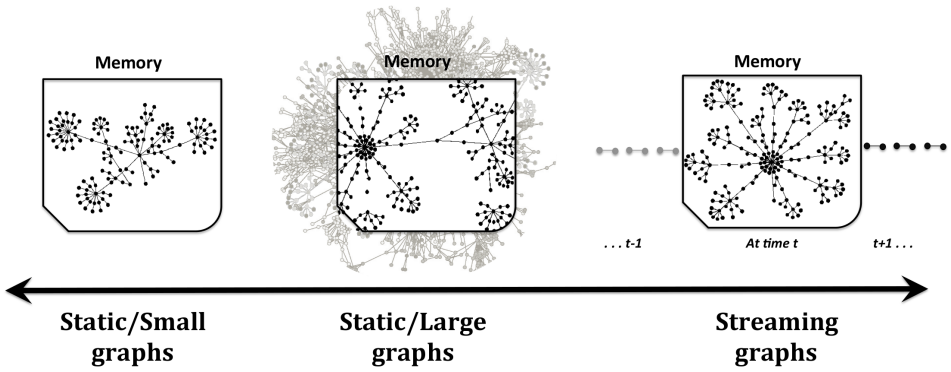
Metody pro vzorkování se mohou dělit podle toho, na jaký typ sítě z hlediska dostupnosti je lze použít. Rozsáhlou síť někdy nelze uložit celou do paměti a je k ní pouze omezený přístup prostřednictvím několika málo vrcholů a sousedů těchto vrcholů. Následující možnosti přístupu ke grafu je nutné při vzorkování zohlednit [21]:

- **Plný přístup ke grafu** - celá rozsáhlá síť je viditelná a uložená v paměti. Je možný náhodný výběr vrcholu nebo hrany z celé sítě. Pro graf s plným přístupem je možné použít metody založené na náhodném výběru vrcholů nebo náhodném výběru hran, případně kombinací obou metod.

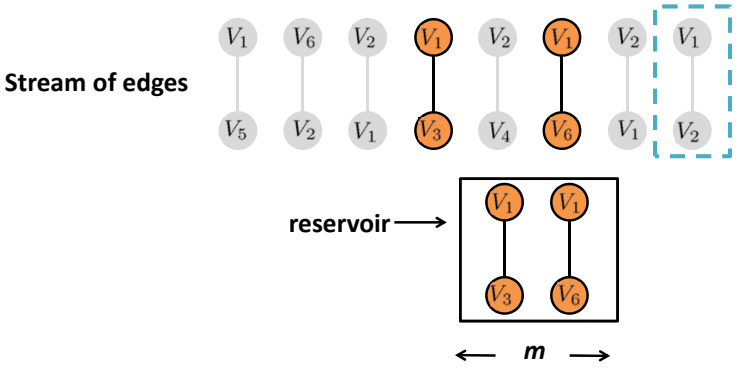
- **Omezený přístup ke grafu** - síť je skryta, nicméně umožňuje procházení vrchol po vrcholu a prozkoumávání sousedů aktuálního vrcholu. Předpokladem je, že síť je propojená do jedné komponenty. Tento způsob je vhodný v případě velmi rozsáhlé sítě, která se nevejde do hlavní paměti. Metody vhodné pro tento způsob jsou založeny na procházení grafu - Forest Fire, prohledávání do šířky nebo prohledávání do hloubky. Další metody jsou založeny na náhodných procházkách - Random Walk, Random Jump nebo Metropolis-Hastings Random Walk. Během procházení grafu jsou navštívené vrcholy ukládány jako vzorek původního grafu.

- **Stream dat** - se používá tam, kde je omezena hlavní paměť a data se rychle přesouvají. Tento způsob je vhodný pro dynamické sítě. Hrany přicházejí ke zpracování buď v určeném pořadí, nebo hrany incidentní s jedním vrcholem přicházejí spolu. Stream hran je tak masivní, že není možné všechna data uložit do hlavní paměti. U tohoto způsobu je důležité

efektivní zpracování v reálném čase. Většina algoritmů pro vzorkování streamovaných grafů je založena na náhodném ukládání do paměti pevné velikosti. Každá přicházející hrana je s určitou pravděpodobností vybrána a uložena do kontejneru pevné velikosti. Pokud je kontejner plný, nová hrana nahradí již dříve uloženou starší hranu. Uložené hrany na konci představují redukovaný graf [21]. Postup vzorkování streamovaného grafu je znázorněn na obrázku 4. Na obrázku 3 je znázornění způsobu práce s různě velkými grafy a vzorkování podle velikosti a dostupnosti grafu.



Obrázek 3: Práce s grafy a vzorkování podle velikosti a dostupnosti grafu [21].



Obrázek 4: Znázornění streamu grafu [21].

Experimenty v této práci se prováděly nad sítěmi s plným přístupem a používaly se algoritmy vhodné jak pro plný přístup, tak i pro omezený přístup ke grafu. Algoritmy pro vzorkování streamovaných grafů se tato práce nezabývala.

3.5 Metody pro vzorkování rozsáhlých sítí

Je mnoho způsobů, jak získat vzorek z rozsáhlé sítě. Obecně je lze rozdělit do dvou tříd podle toho, zda je možný plný přístup k síti či nikoli (jak bylo popsáno v předešlé kapitole 3.4). Jestliže hlavní paměť pojme celou síť, je možné vybrat náhodný vrchol nebo hranu. Tyto metody jsou

založené na náhodném výběru vrcholů nebo **náhodném výběru hran**. Jinak je nutné použít techniky **založené na procházení grafu**.

3.5.1 Metody založené na náhodném výběru vrcholů

Random Node

Nejjednodušší cestou k získání vzorku je náhodně vybrat podmnožinu vrcholů z původního grafu pomocí algoritmu Random Node (RN). Nejdříve je s uniformní pravděpodobností vybrán požadovaný počet vrcholů $V_S \subseteq V$. Jestliže je velikost vzorku stanovena například na 15% původní sítě, bude každý vrchol vybrán s pravděpodobností $p = 0.15$. Následně se do vzorku přidají hrany $E_S = \{(u, v) \in E | u \in V_S, v \in V_S\}$, tedy z původního grafu jsou ponechány pouze hrany mezi vrcholy V_S ze vzorku.

Předpokladem pro RN algoritmus je plně přístupná síť. Nevýhodou je, že vzorek získaný touto metodou moc dobře nerespektuje distribuce stupňů původní sítě dle mocninného zákona [19]. Pseudokód pro RN metodu je uveden ve výpisu 1.

Algorithm 1 Random node (p, graph)

```
1: V_s ← ∅, E_s ← ∅
2: V ← graph.nodes, E ← graph.edges
3: while |V_s| < (p · |V|) do                          ▷ add nodes
4:     n ← random(V)                                   ▷ uniformy random from V
5:     V_s ← V_s ∪ {n}
6: for i ← 0 to |E| do                                 ▷ add edges
7:     (u, v) ← E[i]
8:     if u ∈ V_s and v ∈ V_s then
9:         E_s ← E_s ∪ {(u, v)}
   return G_s(V_s, E_s)
```

Random Degree Node

Metoda Random Degree Node (RDN) je založena na výběru vrcholů proporčně vůči stupni daného vrcholu. Pravděpodobnost, že je vrchol vybrán, závisí na stupni vrcholu. Čím větší stupeň vrchol má, tím je větší šance, že bude ve vzorku. Proto i tato metoda příliš nerespektuje distribuce stupňů původní sítě dle mocninného zákona. Ve vzorku je příliš mnoho vrcholů s vysokým stupněm a lze tak očekávat, že výsledný graf bude velmi hustý [2].

Další podobnou metodou je Random Pagerank Node (RPN), která provádí výběr vrcholů s pravděpodobností úměrnou jejich PageRanku. Metoda u orientovaných grafů dobře zachovává tvar distribuce vstupních stupňů a hot-plot distribuci.

3.5.2 Metody založené na náhodném výběrů hran

Random Edge

Metoda Random Edge (RE) provádí výběr hran s uniformní pravděpodobností a přidává je do vzorku $E_S \subseteq E$ tak dlouho, dokud vzorek není dostatečně velký. Na rozdíl od náhodného výběru vrcholů, RE metoda nemění četnost hran (vzhledem k měřítku), protože výběr hrany připojené k vrcholu i je závislý na jeho stupni k_i. Vrchol se stupněm k v původním grafu bude mít stupeň pk ve vzorku, kde p je pravděpodobnost výběru jedné hrany. To má za následek stejnou distribuci stupňů jako v původním grafu [19].

S metodou RE se váže několik nevýhod. Je-li počet hran původního grafu $|E|$, bude ve vzorku $p|E|$ hran rozloženo mezi n vrcholů, což má za následek nízký průměrný stupeň vrcholu, Jestliže průměrný stupeň klesne pod hodnotu 1, lze očekávat, že vzorek bude postrádat jednu hlavní velkou komponentu a bude obsahovat mnoho malých komponent. Redukovaný graf bude velmi řídce propojen a komunitní struktura nebude zachována [2].

Hybridní algoritmus

Hybridní algoritmus (HYB) kombinuje předešlou metodu RE s metodou Random Node-Edge (RNE). Metoda RNE je mírnou variací náhodného výběru hran. Nejprve se s uniformní pravděpodobností vybere vrchol, následně se vybere náhodně hrana s ním incidentních. Metoda Random Edge mírně zvýhodňuje vrcholy s vyšším stupněm, protože je s nimi incidentní více hran. Random Node-Edge naopak tímto neduhem netrpí.

Algorithm 2 Hybridní algoritmus (p, c, graph)

```
1:  V_s ← ∅, E_s ← ∅
2:  V ← graph.nodes, E ← graph.edges
3:  while |V_s| < (p · |V|) do
4:      if random(0, 1) < c then                          ▷ Random Node-Edge
5:          u ← random(V)                                 ▷ get random node
6:          k_u ← neighbours(u)
7:          v ← random(k_u)                               ▷ get random neighbour of node u
8:          V_s ← V_s ∪ {u, v}
9:          E_s ← E_s ∪ {(u, v)}
10:     else                                              ▷ Random Edge
11:         (u, v) ← random(E)
12:         V_s ← V_s ∪ {u, v}
13:         E_s ← E_s ∪ {(u, v)}
    return G_s(V_s, E_s)
```

Hybridní algoritmus kombinuje uvedené metody do jedné. S pravděpodobností c se provede krok RNE, s pravděpodobností $1-c$ se provede krok RE. Podle [2] dosáhla tato metoda nejlepších výsledků s parametrem c, který má hodnotu 0,8. Metody založené na náhodném výběru vrcholů umí dobře zachovávat tvar distribuce slabě souvislých komponent, ale v průměru pro všechny

porovnávané distribuce podávají nejhorší výsledky. Pseudokód Hybridní metody je uveden ve výpisu 2.

3.5.3 Metody založené na procházení grafu

Random Walk

Náhodná procházka je algoritmus, u kterého je následující navštívený vrchol volen zcela náhodně z množiny sousedních vrcholů aktuálního vrcholu. Je to proces, který začíná ve vrcholu v_0 a po k-tém kroku skončí ve vrcholu v_k. V každém kroku k je pravděpodobnost přechodu z vrcholu v_k do sousedního vrcholu dána vztahem 14, kde k_{v_k} je stupeň vrcholu v_k.

$$p = \frac{1}{k_{v_k}} \tag{14}$$

Počáteční vrchol v_0 je vybrán z vektoru P_0, který každému vrcholu přiřazuje pravděpodobnost, s jakou v něm bude náhodná procházka započata. P_k je vektor rozložení pravděpodobnosti, s jakou se bude náhodná procházka nacházet v daných vrcholech po k krocích, jestliže bylo počáteční rozložení P_0. Pravděpodobnost přechodu je určena maticí pravděpodobnosti přechodu M_{uv}, kde $u, v \in V$. Matice pravděpodobnosti přechodu je definována následovně:

$$M_{uv} = \begin{cases} \frac{1}{k_u} & \text{pokud } \{u, v\} \in E \\ 0 & \text{jinak} \end{cases} \tag{15}$$

Pravděpodobnost přechodu v kroku $k + 1$ závisí na předchozí pozici k a na matici pravděpodobnosti přechodu a lze ji vyjádřit jako:

$$P_{k+1} = M^T P_k \tag{16}$$

Metoda Random Walk v prvním kroku vybere s uniformní pravděpodobností počáteční vrchol v_0, ze kterého je simulována náhodná procházka. V každém kroku k se vybere jeden vrchol u z množiny sousedů vrcholu v_{k-1}. Výběr se provádí s uniformní pravděpodobností podle matice pravděpodobnosti uvedené výše nebo pravděpodobnost výběru může záviset například na vahách jednotlivých hran. Nechť je další vrchol $v_k \leftarrow u$ a do vzorku se uloží hrana $(v_{(k-1)}, v_k)$. Kroky se opakují, dokud nemá vzorek požadovanou velikost. V každém kroku se s pravděpodobností c algoritmus vrátí na počáteční vrchol v_0 a začne novou cestu. Obvyklá hodnota této pravděpodobnosti je $c = 0,15$ [2]. Postup průchodu sítí je uveden v pseudokódu 3.

Výsledný vzorek je tvořen jen z jedné komponenty. Jestliže počáteční vrchol v_0 leží v malé izolované komponentě, může nastat problém a vzorek nemusí dosáhnout požadované velikosti. Je tedy dobré kontrolovat velikost vzorku v každém kroku a pokud po dostatečném počtu kroků (např. $100 \cdot N$, kde N je počet vrcholů) nemá vzorek požadovanou velikost, algoritmus se restartuje a vybere se jiný počáteční vrchol. Redukovaný graf vytvořený Random Walk algoritmem zachovává tvar distribuce vstupních stupňů, singulárních hodnot a prvního levého singulárního

vektoru. Pravděpodobnost, že vrchol u bude ve vzorku, je dána vztahem 17, kde k_u je stupeň vrcholu u a m je počet hran grafu. Vrcholy s vyšším stupněm mají tedy vyšší šanci na výběr [21].

$$p_k = \frac{k_u}{2m} \tag{17}$$

Algorithm 3 Random Walk (p, c, graph)

```
1:  V_s ← ∅, E_s ← ∅
2:  startNode, u ← random(V)                          ▷ initial node
3:  while |V_s| < (p · |V|) do
4:      if random(0, 1) < c then
5:          u ← startNode                              ▷ restart walk
6:      else
7:          K_u ← neighbours(u)
8:          v ← random(K_u)                            ▷ get random neighbour of node u
9:          V_s ← V_s ∪ {u, v}
10:         E_s ← E_s ∪ {(u, v)}
11:         u ← v
    return G_s(V_s, E_s)
```

Random Jump

Metoda Random Jump pracuje podobně jako Random Walk. Jediným rozdílem je, že s pravděpodobností c se algoritmus nevrátí na počáteční místo, ale náhodně vybere jakýkoliv jiný vrchol $v \in V$, ze kterého pokračuje v procházení. Tato metoda nemá, na rozdíl od Random Walk metody, problémy s uváznutím v malé izolované komponentě. Výchozí hodnota pro c je opět udávána jako $0,15$. Random Jump metoda upřednostňuje vrcholy s vysokým stupněm a distribuce stupňů tak není zachována [22].

Forest Fire

Metoda Forest Fire vychází z **Forest Fire modelu**, který generuje grafy, kterým s postupem času stoupá hustota a klesá průměr [23]. Model zachycuje důležité pozorování z reálných sítí a je například schopen generovat grafy, které mají i distribuce výstupních stupňů ve tvaru mocninné křivky.

Forest Fire algoritmus je kombinací Showball sampling metody a Random Walk metody. Showball metoda je založena na rekurzivním výběru všech sousedů již vybraných vrcholů tak, že na počáteční vrchol se jako na sněhovou kouli postupně „nabaluji“ sousedé, sousedé sousedů a tak dál. Forest fire algoritmus začíná výběrem náhodného vrcholu v_0 a přidáním vrcholu v_0 do nově vytvořeného vzorku. Následně se začne „zapalovat“ část hran vrcholu v_0 a vrcholy s nimi incidentní. Proces se rekurzivně opakuje pro každý „zapálený“ vrchol. Pokud je vrchol zapálen,

tak část jeho výstupních hran je opět zapálena. Původní algoritmus pracuje s orientovaným grafem a má dva parametry (viz další odstavec). Pro neorientované grafy je použit jen parametr p_f (pseudokód je k dispozici ve výpisu 4).

Algorithm 4 Forest Fire (p, p_f, $graph$)

```
1:  V_s ← ∅, E_s ← ∅
2:  Q ← queue
3:  u ← random(V)                                              ▷ initial node
4:  V_s ← V_s ∪ {u}
5:  Q.enqueue(u)
6:  while |V_s| < (p · |V|) and not Q.empty() do
7:      u ← Q.dequeue()
8:      K_u ← neighbours(u)
9:      n ← Geom(1 / (1 + p_f/(1−p_f)))                        ▷ geometric distribution with mean p_f/(1−p_f)
10:     N ← random(K_u, n)        ▷ get n random neighbours of node u; if u < |K_u| then get all
11:     for i ← 0 to |N| do
12:         v ← N[i]
13:         if v not exist in V_s then
14:             V_s ← V_s ∪ {v}
15:             E_s ← E_s ∪ {(u, v)}
16:             Q.enqueue(v)
    return G_s(V_s, E_s)
```

Počet spálených sousedů k je náhodné číslo generované z geometrického rozdělení $K \sim Geom(p)$ s průměrem $\bar{x}$ [24]. Průměr $\bar{x}$ je vypočítán pomocí vzorce 18.

$$\bar{x} = \frac{p_f}{1 - p_f} \tag{18}$$

Autoři modelu doporučují hodnotu $p_f = 0,7$, což znamená, že každý vrchol spálí v průměru 2,33 sousedů. Parametr p_f je **dopředná pravděpodobnost zapálení** (forward burning probability) a určuje počet zapálených výstupních hran. V případě orientovaných grafů algoritmus ještě pracuje s parametrem **zpětná pravděpodobnost zapálení** (backward burning probability) p_b, která určuje počet zapálených vstupních hran. Z vrcholu v se do vzorku přidá x výstupních hran $\{v, v_x\}$ a y vstupních hran $\{v_y, v\}$ a vrcholy v_x, v_y s těmito hranami incidentní, které ještě nebyly navštíveny. V případě, že vrchol v nemá dostatečný počet incidentních hran, které je potřebné spálit, vyberou se všechny hrany.

Proces se opakuje tak dlouho, dokud nebylo spáleno dostatečné množství vrcholů a vzorek tak nemá požadovanou velikost. Na Forest Fire metodu lze nahlížet jako na pravděpodobností verzi Breadth-first search metody, kdy každý soused aktuálního vrcholu je navštíven s pravděpodobností p. Pro Breadth-first search algoritmus je pravděpodobnost $p = 1$. Proto je u Forest Fire metody šance, že algoritmus skončí dřív, než se vybere dostatečný počet vrcholů. Tato metoda dobře zachovává tvar distribuce vstupních stupňů, hop-plot, prvního levého singulárního vektoru

i singulárních hodnot. Metoda vykazuje velmi dobré výsledky zejména pro back-in-time využití vzorků.

Metropolis-Hastings Random Walk

Dosavadní metody založené na náhodných procházkách upřednostňují vrcholy s vysokým stupněm a distribuce stupňů původního grafu není zachována. Je možné zajistit to, aby algoritmus náhodné procházky navštívil každý vrchol s uniformní pravděpodobností za předpokladu, že není znám celý graf a v každém kroku je dosažitelný pouze aktuální vrchol a jeho sousedé? Řešením je Metropolisův-Hastingsův algoritmus v kombinaci s náhodnou procházkou. **Metropolisův-Hastingsův algoritmus** je metoda typu Markov chain Monte Carlo pro získávání posloupností náhodných vzorků z nějakého pravděpodobnostního rozdělení.

Markovův řetězec v tomto případě představuje posloupnost navštívených vrcholů, které mají požadovanou distribuci stupňů. Klíčovou myšlenkou je nahrazení navštívených vrcholů jinými vrcholy, které svým stupněm budou lépe odpovídat původní distribuci stupňů. U klasické Random Walk metody jsou vrcholy vzorkovány s nerovnoměrnou distribucí definovanou vztahem 19, kde k_u je stupeň vrcholu u. MHRW naproti tomu zajistí vzorkování vrcholů se stejnou pravděpodobností 20. To je dosaženo přechodovou maticí, která je uvedena ve vztahu 21. [25]

$$\pi_u^{RW} \sim k_u \tag{19}$$

$$\pi_u = \frac{1}{|V|} \tag{20}$$

$$P_{u,v}^{MH} = \begin{cases} \frac{1}{k_u} \cdot min(1, \frac{k_u}{k_v}) & \text{pokud v je soused u} \\ 1 - \sum_{y \neq u} P_{u,y}^{MH} & \text{když v = u} \\ 0 & \text{jinak} \end{cases} \tag{21}$$

Pseudokód této metody je uveden ve výpisu 5. Na začátku algoritmu se s uniformní pravděpodobností vybere počáteční vrchol v s nenulovým stupněm. Následně se ze sousedů vrcholu v vybere jeden vrchol u a poté se vygeneruje náhodné číslo p z uniformní distribuce $U(0, 1)$. Pokud pro náhodné číslo p platí vztah 22, navržený vrchol je akceptován a algoritmus se přesune do vrcholu u. Jinak zůstane v původním vrcholu v.

$$p \leq \frac{k_v}{k_u} \tag{22}$$

Jestliže je stupeň vrcholu u (k_u) malý, vrchol u bude vybrán ze seznamu sousedů v jako kandidát jen s malou pravděpodobností. Ale pokud se tak stane, s velkou pravděpodobností bude návrh přijat a algoritmus se přesune do vrcholu u. Vrcholy s malým stupněm jsou akceptovány

a některé vrcholy s velkým stupněm jsou zamítány. Výsledkem je eliminace upřednostňování vrcholů s velkým stupněm [26].

Algorithm 5 Metropolis-Hastings Random Walk (p, graph)

```
1: V_s ← ∅, E_s ← ∅
2: u ← random(V)                                   ▷ initial node
3: while |V_s| < (p · |V|) do
4:     K_u ← neighbours(u)
5:     v ← random(K_u)          ▷ select node w uniformly at random from neighbours of u
6:     r ← random(0,1)                ▷ generate uniformly at random a number 0 ≤ r ≤ 1
7:     K_v ← neighbours(v)
8:     if r ≤ |K_u| / |K_v| then
9:         V_s ← V_s ∪ {u}
10:        E_s ← E_s ∪ {(u, v)}
11:        u ← v
12:    else
13:        stay at u
   return G_s(V_s, E_s)
```

Topologically Divided Stratums

Zkoumané vlastnosti sítě popsané v kapitole 5.1 nezohledňují jednu důležitou vlastnost - topologickou strukturu sítí. Topologická struktura zachycuje skutečné a logické podoby a formuje síť do určitých tvarů. Redukovaný graf by měl mít podobnou strukturu jako původní graf. Předešlé metody vzorkování na tuto vlastnost nebyly zaměřeny.

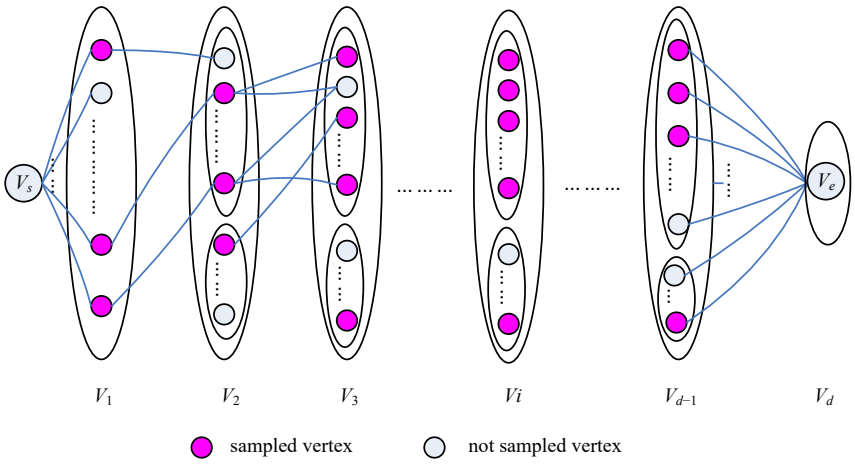
Algoritmus Topologically Divided Stratums pracuje s průměrem d původního grafu. Na začátku se vybere jeden ze dvou od sebe nejvzdálenějších vrcholů původního grafu se vzdáleností d. Následně se vrcholy původního grafu rozdělí do podmnožin podle vzdálenosti od počátečního vrcholu. Z každé podmnožiny je vybráno p procent vrcholů, kde p představuje požadovanou velikost vzorku. Cílem je vybrat vrcholy a hrany rovnoměrně napříč celým původním grafem.

Pseudokód DS metody je uveden ve výpisu 6. Z původního grafu G jsou zjištěny dva nejvzdálenější vrcholy s nejdelší cestou o velikosti d - průměrem grafu. Jeden z těchto vrcholů je zvolen jako výchozí vrchol v_s a je přidán do množiny vrcholů vzorku V_S. Množina vrcholů původního grafu V_G je rozdělena do d podmnožin $\{V_1, V_2, ..., V_d\}$ podle vzdálenosti daného vrcholu v_i od v_s. Podmnožina V_i reprezentuje vrcholy, které jsou od vrcholu v_s na vzdálenost i. Před samotným výběrem vrcholů je podmnožina V_i dál rozdělená do dvou podmnožin V_{i_adj} a V_{i_cpl}. Vrchol v množině V_{i_adj} má alespoň jednu hranu incidentní s nějakým vrcholem z množiny S_{i-1}, kde množina S_{i-1} představuje již vybrané vrcholy z množiny V_{i-1}. Vrcholy z množiny V_{i_cpl} nemají žádné spojení s vrcholy z množiny S_{i-1}.

Následuje samotný výběr vrcholů. Při výběru z množiny V_i je k procent (jak parametr k ovlivňuje úspěšnost metody je v kapitole 5.6) vrcholů vybráno z množiny V_{i_adj} a zbytek je

Algorithm 6 Divided Stratums (p, k, graph)

$V_s \leftarrow \emptyset, E_s \leftarrow \emptyset$
$d \leftarrow$ graph.diameter ▷ diameter of graph
$v_s \leftarrow$ graph.endpoint ▷ randomly pick one endopoint
$V_{s\_0}, V_{s\_1} \cdots V_{s\_d} \leftarrow \emptyset$
for $j \leftarrow 0$ to $|V|$ **do** ▷ Split V to d subsets according to the distance to v_s
    $i \leftarrow \text{distance}(v_s, V[j])$
    $V_i \leftarrow V_i \cup \{V[j]\}$
$V_{s\_0} \leftarrow V_{s\_0} \cup \{v_s\}$
for $i \leftarrow 1$ to d **do**
    for $j \leftarrow 0$ to $|V_i|$ **do** ▷ split V_i to V_{i_adj} and V_{i_cpl}
        if exist link between $V_i[j]$ and some node $\in V_{s\_i-1}$ **then**
            $V_{i\_adj} = V_{i\_adj} \cup V_i[j]$
        else
            $V_{i\_cpl} = V_{i\_cpl} \cup V_i[j]$
    randomly pick $k \cdot p$ percentage nodes in V_{i_adj} to V_{s_i}
    randomly pick $(1-k) \cdot p$ percentage nodes in V_{i_cpl} to V_{s_i}
$V_s \leftarrow V_{s\_0} \cup V_{s\_1} \cdots V_{s\_d}$
for $i \leftarrow 0$ to $|E|$ **do** ▷ add edges
    $(u, v) \leftarrow E[i]$
    if $u \in V_s$ **and** $v \in V_s$ **then**
        $E_s \leftarrow E_s \cup \{(u, v)\}$
return $G_s(V_s, E_s)$



Obrázek 5: Proces výběru vrcholů metody Topologically Divided Stratums.

vybrán z V_{i_cpl}. Výběr vrcholů na základě propojenosti s již vzorkovaným vrcholem (z množiny V_{i_adj}) zachovává souvislost vzorku. Na druhou stranu je také potřebné vybrat i odlehlejší, nepropojené vrcholy (z množiny V_{i_cpl}) tak, aby vzorek zachovával topologickou strukturu. Proces výběru z množin je znázorněn na obrázku 5. Po výběru vrcholů ze všech množin V_x jsou do vzorku přidány hrany, které se mezi vybranými vrcholy nacházejí v původním grafu [27].

Nevýhodou této metody je nutnost zjištění délky všech nejkratších cest mezi všemi vrcholy původního grafu. To může být výpočetně náročné a metoda tak postrádá smysl. Řešením je zjištění pouze přibližné hodnoty průměru grafu pomocí několika prohledávání grafu do šířky.

Multi-dimenzionální Random Walk

Hlavním problémem metody Random Walk je hrozba uváznutí v malé izolované komponentě nebo lokální hustě propojené oblasti. Úspěšnost metody závisí na počátečním výběru vrcholu a výsledky mohou být velmi odlišné při výběru jiného počátečního vrcholu. Řešením by mohlo být spuštění m nezávislých náhodných procházek (Multiple Independent Random Walk), každá s jiným počátečním vrcholem. Ukázalo se, že tento přístup negeneruje lepší vzorky a dokonce snižuje přesnost odhadu vlastností ze vzorku [28].

Algorithm 7 Frontier Sampling (p, m, graph)

$V_s \leftarrow \emptyset, E_s \leftarrow \emptyset$
$V \leftarrow$ graph.nodes
$L \leftarrow \text{random}(V, m)$ ▷ Select m nodes uniformly at random
while $|V_s| < (p \cdot |V|)$ **do**
    $u_i \leftarrow \text{randomDegree}(L)$ ▷ Select u_i with probability $\frac{k_{u_i}}{\sum_{v \in L} k_v}$
    $K_{u_i} \leftarrow \text{neighbours}(u_i)$
    $v \leftarrow \text{random}(K_{u_i})$ ▷ Select random neighbour of u
    $V_s \leftarrow V_s \cup \{u_i, v\}$
    $E_s \leftarrow E_s \cup \{(u_i, v)\}$
    $L \leftarrow (u_1, \ldots, u_{i-1}, v, u_{i+1}, \ldots, u_m)$ ▷ Replace u by v in L
return $G_s(V_s, E_s)$

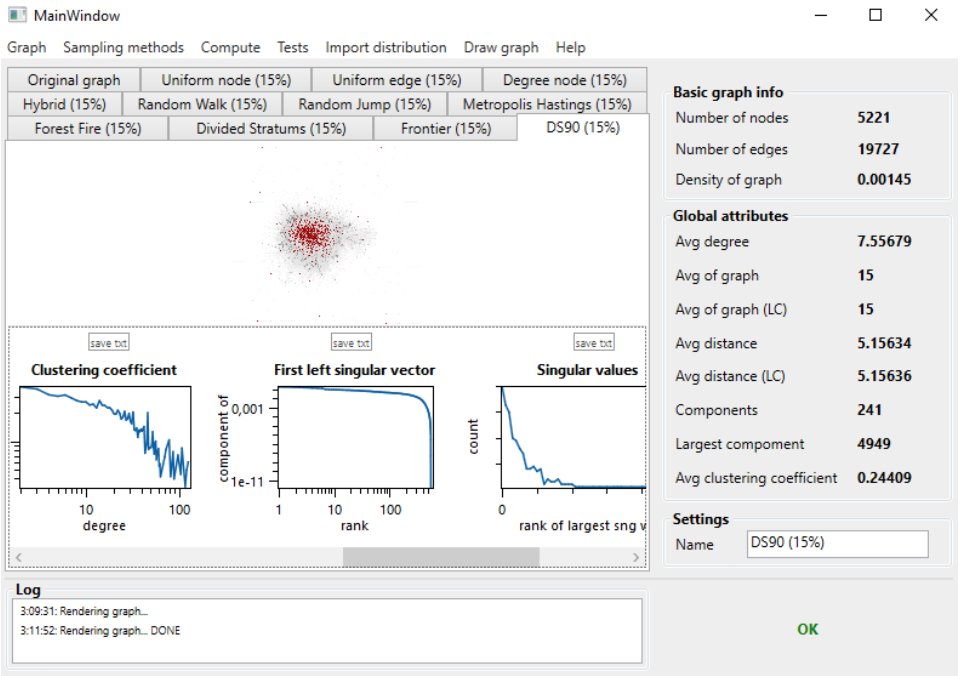
Z toho důvodu byla navržena metoda Multi-dimenzionální Random Walk, které se také říká Frontier Sampling. m-dimenzionální Random Walk provádí m závislých náhodných procházek. V prvním kroku se vybere množina L počátečních vrcholů, kde $|L| = m$. Poté se v každém kroku vybere z L jeden vrchol v s pravděpodobností úměrnou stupni vrcholu danou vztahem 23.

$$p_v = \frac{k_v}{\sum_{u \in L} k_u} \tag{23}$$

Vrchol u je náhodně s uniformní pravděpodobností vybrán z množiny sousedů vrcholu v a hrana (v, u) je přidána do vzorku. Vrchol u následně nahradí vrchol v v množině L. To se opakuje, dokud nemá vzorek požadovanou velikost. Celý postup je uveden v pseudokódu 7. Frontier sampling metoda velmi dobře zachovává distribuci stupňů vrcholů a distribuci shlukovacího koeficientu [26].

4 Implementace

Aplikace pro experimenty s redukovanými sítěmi je implementována v *.NET frameworku* verzi 4.6 v jazyce C# . Jako vývojové prostředí bylo použito *Visual Studio* ve verzi 2015 (14.0). Implementace obsahuje knihovnu s třídami, zajišťující práci se sítěmi, jejich vzorkování a výpočet vlastností. Nad tím je postavené grafické uživatelské rozhraní pomocí *WPF*, umožňující práci v GUI.



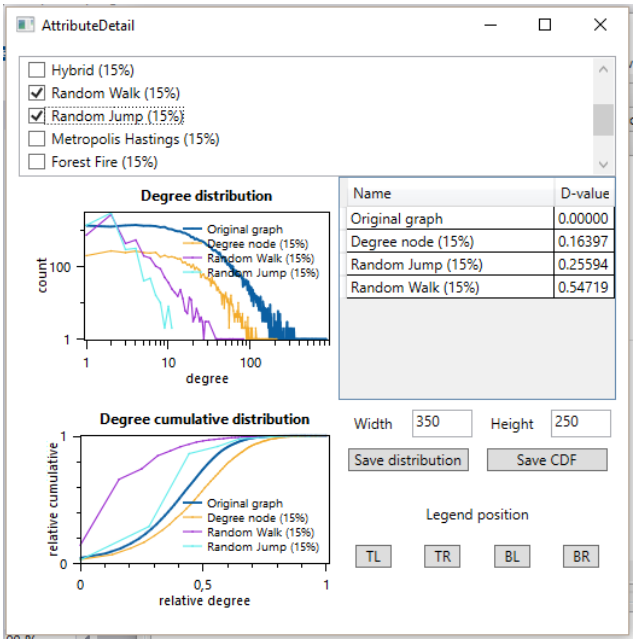
Obrázek 6: Ukázka prostředí pro testování vzorků - hlavní okno

Pro maticové výpočty byla použita knihovna *Math.NET Numerics* ve verzi 3.17[2]. Je použita pro výpočet prvního levého singulárního vektoru pomocí SVD rozkladu matice sousednosti. Pro vykreslení distribucí byla použita knihovna *OxyPlot* ve verzi 1.0[3]. Zajišťuje grafickou reprezentaci vlastností sítě v podobě spojnicových grafů a jejich vizuální porovnávání. Výstupy této knihovny jsou součástí této práce v kapitole 5. Knihovna *Graphviz* ve verzi 2.28[4] je použita k vizualizaci menších sítí.

Program je ovládán pomocí grafického uživatelského rozhraní. Umožňuje generování náhodných sítí, sítí malého světa a bezškálových sítí. Rovněž umožňuje načtení sítí ze souboru. U každé zkoumané sítě jsou přehledně zobrazeny vypočtené globální vlastnosti a série obrázků s distribucemi, které jsou popsané v kapitole 5.1. Po vytvoření vzorku z původní sítě se otevře nová karta - opět s přehledem vlastností, které lze srovnávat s původní sítí. V detailu vybrané

[2] `https://numerics.mathdotnet.com/`
[3] `http://www.oxyplot.org/`
[4] `http://www.graphviz.org/`

distribuce lze srovnávat kumulativní distribuční funkci a D-hodnoty dané distribuce a to pro různé metody. Ukázka prostředí aplikace je na obrázku 6 a 7. Aplikace umožňuje export vytvořených sítí do souborů k dalšímu zpracování a export vypočtených distribucí. Dál umožňuje import distribucí vypočtených v jiných nástrojích, například pomocí jazyku R[5].



Obrázek 7: Ukázka prostředí pro testování vzorků - porovnání distribuce

4.1 Popis funkčnosti

Nejprve je potřebné načíst počáteční síť ze souboru. Podporovaný typ souboru je CSV s možností zvolení typu oddělovače. Alternativně lze síť vygenerovat pomocí tří modelů - Barabási–Albert, Watts-Strogatz nebo Erdős–Rényi. Poté již lze generovat vzorky zvolením metody a velikosti požadovaného vzorku, případně dalších parametrů pro danou metodu. Mezi původní sítí a vzorky lze přepínat, na pravé straně se nacházejí vypočtené globální vlastnosti aktuálně zobrazené sítě. Pokud je nějaká vlastnost nulová, je potřebné provést výpočty v menu.

V dolní části se nacházejí distribuce pro aktuálně zobrazenou síť, kterou lze uložit do textového souboru pro další zpracování. Po dvojkliku na distribuci se zobrazí distribuce jedné vlastnosti původní sítě a všech vzorků. Je zobrazen obrázek distribuce, normalizovaná relativní kumulativní funkce a D-hodnoty pro jednotlivé vzorky. Tím je možné porovnávat distribuci jedné vlastnosti napříč všemi vzorky (viz obrázek 7). Celý projekt lze uložit do binárního souboru a znovu otevřít, není tak nutné opakovaně počítat vlastnosti sítě. K dispozici je možnost uložení aktuálně zobrazeného vzorku do souboru v podobě seznamu hran nebo ve formátu *GraphML*[6].

[5] `https://cran.r-project.org/`
[6] `http://graphml.graphdrawing.org/`

GraphML je formát pro ukládání grafů, který je založen na XML formátu. Je podporován programem pro vizualizaci sítí *Graphviz*[7] a jazykem R, který je využit pro dodatečné výpočty některých vlastností.

4.1.1 Popis parametrů jednotlivých metod

- Pro každou metodu je nutné nastavit požadovanou velikost vzorku v procentech původní velikosti grafu.
- **Hybridní algoritmus** - pravděpodobnost p v procentech, s jakou se provede krok RNE. S pravděpodobností $p - 1$ se provede krok RE.
- **Random Walk** - pravděpodobnost c, s jakou se v každém kroku vrátí algoritmus do počátečního vrcholu.
- **Random Jump** - pravděpodobnost c, s jakou se v každém kroku algoritmus přesune na jiný náhodně vybraný vrchol.
- **Forest Fire** - parametr p_f, který je použit při výpočtu průměru geometrického rozdělení. Z tohoto rozdělení je získáno číslo představující počet spálených sousedů.
- **Divided Stratums** - c procento vrcholů, které je vybráno z množiny V_{i_adj}.
- **Frontier Sampling** - parametr m udávající počet počátečních, náhodně vybraných vrcholů.

4.2 Diagram tříd

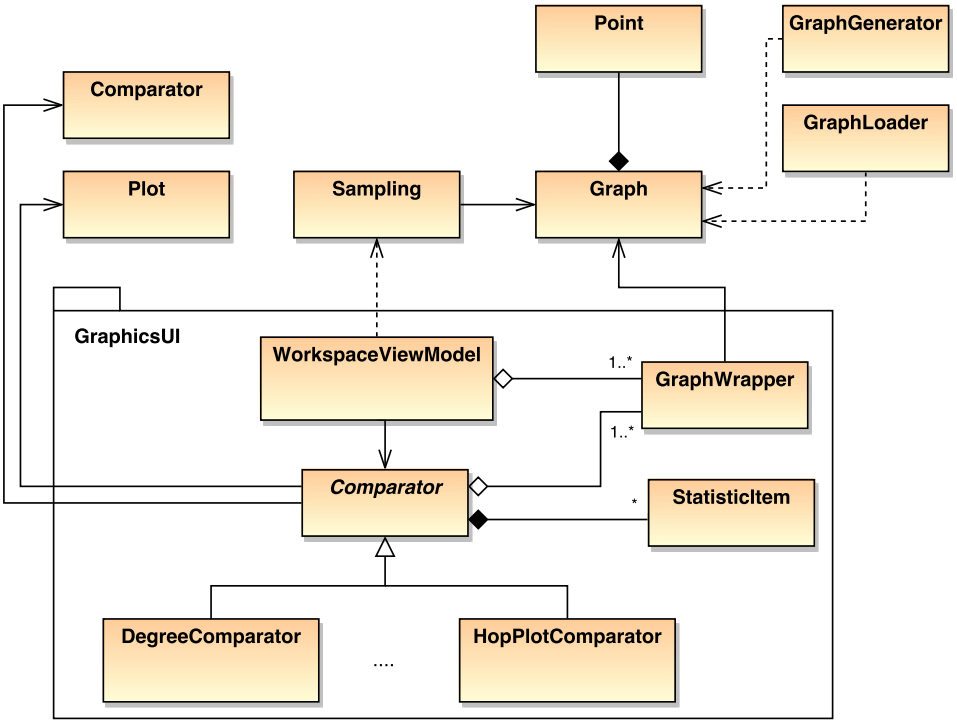
4.2.1 Graph

`Graph` je hlavní třída, která reprezentuje síť. Poskytuje přístup k jednotlivým vlastnostem a distribucím grafu. Graf je reprezentován seznamem sousedů a navíc pomocným seznamem hran, který slouží pro export grafu do souboru. Metody pro výpočet vlastností grafu jsou volány pomocí objektu třídy `GraphWrapper`, který obaluje třídu `Graph` a poskytuje dodatečné metody pro práci v grafické nadstavbě (viz diagram tříd 8).

4.2.2 GraphGenerator

Třída `GraphGenerator` poskytuje statické metody pro generování sítí. Je možné generovat náhodné sítě, sítě malého světa a bezškálové sítě. Výstupem je objekt třídy `Graph`.

[7] `https://gephi.org/`



Obrázek 8: Diagram tříd základní části a GUI nadstavby.

4.2.3 GraphLoader

Třída `GraphLoader` načítá síť ze souboru. Obsahuje statickou metodu `loadFromCsv`, která načítá CSV soubory se seznamem hran. Je možné specifikovat oddělovač vrcholů jednotlivých hran. Výstupem je objekt třídy `Graph`.

4.2.4 Sampling

Třída `Sampling` poskytuje celkem 10 algoritmů, která byly popsány v kapitole 3.5 a které vzorkují grafy. Konstruktor třídy přijímá jako parametr objekt typu `Graph`, představující původní graf, ze kterého se budou generovat vzorky. Výstupem metod je opět objekt třídy `Graph`.

4.2.5 Comparator

Třída `Comparator` zajišťuje operace s distribucí původního grafu a vzorku a porovnává je. Na vstupu metody `Compare` jsou dvě distribuce, které se převedou na kumulativní distribuční funkce. Následně je osa x převedena na logaritmickou osu a distribuce jsou převedeny do intervalu $\langle 0, 1 \rangle$ [2]. Výstupem metody `Compare` je D-hodnota, udávající maximální rozdíl mezi distribucí původního grafu a vzorku. Postup normalizace a výpočtu D-hodnoty je uveden v kapitole 5.2.

4.2.6 GraphicsUI.Comparator

Abstraktní třída `Comparator` GUI nadstavby vytváří graf distribuce a kumulativní distribuce. K výpočtům využívá třídu `Comparator`, která byla popsána v odstavci 4.2.5. Pro každou distribuci existuje samostatná třída, která dědí od třídy `Comparator` (například `DegreeComparator` nebo `HopPlotComparator`).

4.2.7 Point

Třída `Point` představuje jeden bod distribuce. Obsahuje komparátor, který porovnává jednotlivé body podle x-ové složky bodu. K ukládání distribuce je použita datová struktura `HashSet<Point>`, která obsahuje metodu `GetViewBetween`. Pomocí této metody se naleznou nejbližší body původní distribuce z levé i pravé strany od bodu p_1, který je z distribuce nějaké vlastnosti vzorku. Vzdálenost od bodu p_1 se měří podle x-ových souřadnic bodu. Následně se s využitím zjištěných nejbližších bodů původní distribuce pomocí lineární interpolace zjistí y-ová souřadnice bodu p_2 původní distribuce v x-ové souřadnici bodu p_1 distribuce vlastnosti vzorku.

4.2.8 Plot

Třída Plot zapouzdřuje knihovnu *OxyPlot*, která slouží ke generování grafů distribucí. Jeden objekt třídy Plot reprezentuje jeden graf. Graf může obsahovat křivku nebo body, přitom může obsahovat i více distribucí.

4.3 Dodatečné výpočty v jazyce R

Vlastní implementace v C# poskytuje výpočet základních globálních vlastností a distribucí uvedených v kapitole 5.1. Pro dodatečné zjištění některých výpočetně náročnějších vlastností a distribucí byl použit jazyk R, protože poskytoval lepší a efektivnější knihovny pro výpočty nad sítí. Součástí implementační částí je i několik R skriptů, které slouží k výpočtu closeness a betweenness centralit, komunitní struktury, asortativity a vlastních čísel matice sousednosti.

Pro práci s grafy byl použit balíček *igraph*, který zajišťuje načtení grafů ve formátu *GraphML*, získání matice sousednosti pro další zpracování a výpočty centralit. Pro SVD rozklad byl použit balíček *rARPACK*, který poskytuje rychlý výpočet vlastních čísel matice a singulárních vektorů. Balíček umožňuje nastavit požadovaný počet n největších vlastních čísel a počet levých nebo pravých singulárních vektorů, čímž se sníží doba výpočtů.

Pro zjištění komunitní struktury byly použity algoritmy Louvain a Infomap. Metoda **Louvain** pomocí hladového algoritmu hledá komunity v síti. Využívá modularitu, která měří, jak dobře je síť rozdělená na komunity. Nejdříve jsou nalezeny malé komunity, které se poté shlukují do větších komunit. Metoda **Infomap** hledá komunity na základě minimalizace délky náhodné procházky [29]. Výsledky výpočtů jsou uložené do textového souboru, který je možné nahrát do hlavní implementace a dál s výsledky pracovat.

5 Experimenty

5.1 Zkoumané vlastnosti sítě

Jedna z klíčových otázek při vzorkování sítí se zabývá tím, jak kvalitní je vzorek. Jak lze změřit kvalitu vzorku a jak určit nejlepší vzorkovací metodu pro daný druh sítě? Při zkoumání kvality vzorků je nutné zohlednit účel daného vzorku. Má mít vzorek podobné (případně škálované vzhledem k velikosti vzorku) vlastnosti jako původní graf? Nebo má mít vzorek podobné vlastnosti, jaké měl původní graf v čase, kdy byl stejné velký jako vzorek? Různé účely vzoru popisuje kapitola 3.3.

Úspěšnost vzorkování je možné určit například prostým porovnáním distribucí některých vlastností sítě. Cílem není nalézt vzorkovací metodu, která bude produkovat dobré vzorky z pohledu jedné konkrétní vlastnosti. Metoda může produkovat vzorky s přesným očekávaným počtem hran nebo očekávaným průměrem sítě, ale další vlastnosti již nemusí odpovídat původní síti. Cílem je identifikovat vzorkovací metody, které vytvářejí co nejpřesnější vzorky z hlediska co největšího počtu vlastností sítě tak, aby mohly vzorky zastupovat původní grafy, například při výpočetně náročných operacích nad grafem.

V následujících odstavcích jsou popsány vlastnosti, ze kterých jsou vytvořeny distribuce vlastností. Tyto distribuce vlastností se porovnávají a slouží k určení podobnosti vzorku a původní sítě. Ukázky distribucí jsou na sérii obrázcích 10. Porovnávané vlastnosti jsou přebrané z práce [2], která zkoumá úspěšnost různých vzorkovacích metod. Distribuce vlastností jsou získány z orientovaných grafů a jsou určeny pro porovnávání sítí pro scale-down účely. V této práci se pracuje pouze s neorientovanými grafy, proto jsou některé distribuce vynechány.

5.1.1 Vstupní stupně

Ze vstupních stupňů jednotlivých vrcholů se vytvoří distribuce vstupních stupňů. Pro každý stupeň d, který je na ose x, se vypočítá počet vrcholů se vstupním stupněm d. Počet výskytů je na ose y. Distribuce vstupních stupňů pro reálné sítě odpovídá mocninnému zákonu a distribuce má dlouhý ocas. Vzorek proto také musí mít podobný tvar distribuce.

5.1.2 Výstupní stupně

Z výstupních stupňů jednotlivých vrcholů se vytvoří distribuce výstupních stupňů. Opět jsou na ose x jednotlivé stupně a na ose y počet výskytů daného stupně. Pro neorientované grafy v experimentální části se tyto dvě vlastnosti nahradí distribucí stupňů vrcholů. Ukázka distribuce vrcholů je na obrázku 9.

5.1.3 Slabě souvislé komponenty

Komponenta je **slabě souvislá**, pokud pro libovolné dva vrcholy u a v v komponentě existuje neorientovaná cesta z u do v [4]. Velikost komponenty je určena počtem vrcholů v komponentě. Z určených velikostí je vytvořena distribuce velikostí slabě souvislých komponent (viz obrázek 10(a)). Na ose x se nachází jednotlivé velikosti a na ose y je počet slabě souvislých komponent dané velikosti.

5.1.4 Silně souvislé komponenty

Dva vrcholy $u, v \in V(G)$, kde $V(G)$ je množina vrcholů komponenty G, jsou spolu silně propojeny, pokud existuje orientovaná cesta jak z u do v, tak i orientovaná cesta z v do u. Komponenta je **silně souvislá**, pokud je každá dvojice vrcholů $u, v \in V(G)$ silně propojená [4]. Každý vrchol je dosažitelný z libovolného vrcholu po orientované cestě. Z určených velikostí je vytvořena distribuce velikostí silně souvislých komponent. V experimentální části práce na neorientovaných grafech tato distribuce nebude použita.

5.1.5 Hop-plot

Formálně lze hop-plot definovat jako: pro každý vrchol u z množiny vrcholů V grafu G se vypočítá počet vrcholů $N_h(u)$ dosažitelných z vrcholu u, které jsou vzdáleny na maximálně h kroků. Pro vzdálenost h se následně provede sečtení počtu vrcholů podle vztahu 24, kde $u \in V$ [30].

$$P(h) = \sum_{u} N_h(u) \tag{24}$$

Vrchol může tvořit pár i se sebou samým, proto pro vzdálenost $h = 0$ se počet párů rovná počtu vrcholů. Pro průměr grafu d, kde $h = d$ je počet párů roven druhé mocnině počtu vrcholů, což je maximální možný počet párů [31].

Hop-plot distribuce kvantitativně určuje konektivitu sítě a vzdálenosti mezi jednotlivými vrcholy. Studuje velikost komunit o určité vzdálenosti a nezabývá se samotnými vzdálenostmi. Hot-plot také slouží jako metrika pro hustotu sítě. Porovnáním hop-plot distribucí dvou grafů lze zjistit, zda mají grafy podobnou konektivitu a komunitní strukturu.

Hop-plot distribuce počtu párů vrcholů, které jsou od sebe vzdáleny h a méně kroků je na obrázku 10(b). Na ose x jsou všechny hodnoty vzdáleností h a na ose y je hodnota $P(h)$, která představuje počet párů se vzájemnou vzdáleností menší nebo rovno h.

5.1.6 Hop-plot na největší slabě souvislé komponentně

Hop-plot distribuce na největší slabě souvislé komponentě, kde na ose x jsou všechny hodnoty vzdáleností h a na ose y je hodnota $P(h)$. Ukázka distribuce je na obrázku 10(c).

5.1.7 Distribuce prvního levého singulárního vektoru matice sousednosti

Při **singulárním rozkladu** (SVD) je původní matice sousednosti M o velikosti $m \times n$ rozložena na unitární matici U o velikosti $m \times m$, diagonální matici Σ o velikosti $m \times n$ a unitární transponovanou matici $V^\star$ o velikosti $n \times n$ (vzorec 25).

$$M = U\Sigma V^\star \tag{25}$$

Levé singulární vektory představují sloupce unitární matice U. První levý singulární vektor se tedy získá jako první sloupec matice U. Prvky vektoru se seřadí podle hodnoty sestupně. Distribuce prvního levého singulárního vektoru je zobrazena na obrázku 10(e). Na ose y jsou jednotlivé hodnoty prvků vektoru, na ose x pořadí hodnot.

5.1.8 Distribuce singulárních hodnot matice sousednosti

V případě symetrické $n \times n$ pozitivně definitní matice (což platí pro matici sousednosti neorientovaných grafů, kterým se tato práce věnuje), se singulární hodnoty získané při SVD rozkladu matice sousednosti rovnají vlastním číslům matice sousednosti.

Spektrální analýza grafu zkoumá vlastní čísla matice sousednosti, které mohou být brány jako „otisk“ daného grafu a definují strukturu grafu. Množina vlastních čísel seřazená vzestupně se nazývá **spektrum grafu**. Z nejvyšší hodnoty vlastního čísla lze například zjistit odolnost sítě vůči šíření viru. Čím menší je nejvyšší hodnota vlastního čísla, tím je síť odolnější [32]. Jedním z cílů spektrální analýzy grafu je odvodit strukturální charakteristiky grafu na základě vlastních čísel matice. Vlastní čísla mohou také charakterizovat modely reálných sítí, určovat komunity v sítích nebo identifikovat hrany, které spojují různé komunity, jenž při odstranění rozdělí graf do několika izolovaných komponent. Vlastní vektor příslušný druhému nejmenšímu vlastnímu číslu Laplaceovy matice se nazývá **Fiedlerův vektor**. Znaménka hodnot prvků Fiedlerova vektoru mohou být použita pro rozdělení grafu na dvě komunity [33].

Studiem topologie internetové sítě byl zjištěn vztah 26 mezi velkými vlastními čísly λ_i a stupněm vrcholu k_i. Proto rozložení velkých hodnot vlastních čísel odpovídá mocninnému rozložení [34].

$$k_i = \lambda_i^2 \tag{26}$$

Singulární hodnoty (vlastní čísla) představují kladná čísla na diagonále diagonální matice Σ, která byla získána pomocí SVD rozkladu matice sousednosti (vzorec 25), seřazená sestupně. Distribuce z těchto čísel je získána tak, že na ose x je pořadí singulárních hodnot (od největší hodnoty) a na ose y je četnost této singulární hodnoty. Ukázka distribuce je na obrázku 10(f).

5.1.9 Shlukovací koeficient

Definice shlukovacího koeficientu je uvedena v kapitole 2.2.4. Pro výpočet shlukovacího koeficientu pro vrchol i je v implementaci použit vzorec 9.

Pro porovnání shlukovacího koeficientu vzorku a původní sítě je vytvořena distribuce průměrného shlukovacího koeficientu C_k pro všechny vrcholy se stupněm k. Ukázka distribuce je na obrázku 10(d). Na ose x jsou jednotlivé stupně vrcholů a na ose y je průměrný shlukovací koeficient C_k pro stupeň k.

5.1.10 Betweenness centralita

Mimo již zmíněné sledované vlastnosti, ze kterých se počítá D-hodnota sledující, jak jsou si dvě distribuce vlastností podobné, se navíc u testovaných sítí v experimentální části sledují další vlastnosti. D-hodnota těchto vlastností se nezapočítává do celkového průměru, který určuje úspěšnost dané metody. Distribuce těchto vlastností původní sítě a vzorku se pouze vizuálně porovnává.

Betweenness centralita vrcholu v je počet nejkratších cest procházejících vrcholem v a vyjadřuje, jak je uzel v potřebný k propojení jiných párů uzlů. Pro vrchol v lze betweenness centralitu určit pomocí vztahu 27, kde g_{st} je počet nejkratších cest mezi vrcholy s, t a n_{st}^v je počet nejkratších cest mezi s a t procházející vrcholem v.

$$x_v = \sum_{st} \frac{n_{st}^v}{g_{st}} \tag{27}$$

Hodnota je nejvyšší, pokud cesty mezi libovolnými dvěma vrcholy sítě vždy procházejí tímto vrcholem. Vrcholy s vysokou hodnotou (mosty, zprostředkovatelé) kontrolují tok informací v síti, nebo se mohou chovat jako úzké hrdlo sítě [35]. Distribuce betweenness centrality se získá podobně jako u shlukovacího koeficientu. Na ose x je stupeň vrcholu a na ose y průměrná hodnota betweenness centrality pro daný stupeň. Ukázka distribuce je na obrázku 10(g).

5.1.11 Closeness centralita

Closeness centralita udává počet vrcholů dělený součtem vzdáleností mezi vrcholem v a všemi ostatními vrcholy (vzorec 28).

$$C_v = \frac{N}{\sum_u d(v,u)} \tag{28}$$

Hodnota je nejvyšší, jestliže z vrcholu v lze dosáhnout ke všem dalším vrcholům v síti přímou nezprostředkovanou vazbou - nejmenší hodnota součtu vzdáleností k ostatním vrcholům. Vrcholy s vysokou closeness centralitou mají velký vliv na to, co se v síti odehrává, protože mají nejrychlejší přístup k celé síti. Proto jsou také náchylné například k virům [35].

Stejně jako u betweenness centrality a shlukovacího koeficientu, na ose x jsou jednotlivé stupně vrcholů sítě a na ose y průměrná closeness centralita pro daný stupeň vrcholu. Ukázka distribuce je na obrázku 10(h).

5.1.12 Distribuce komunit

Jak bylo popsáno v kapitole 2.2.4, reálné sítě obsahují komunity složené z vrcholů, které jsou spolu vzájemně vysoce propojené. Komunity jsou mezi sebou propojené několika slabými vazbami. Distribuce komunit představuje počet komunit dané velikosti. Na ose x se nachází jednotlivé velikosti a na ose y je počet komunit dané velikosti. Ukázka distribuce je na obrázku 10(i). Příslušnost ke komunitě je zjišťována pomocí algoritmu Infomap [29]. Velikost komunity je určena počtem vrcholů v příslušné komunitě.

5.2 Evaluační technika

Cílem experimentů bylo určení metod, které nejlépe zachovávají vlastnosti původní sítě. Jak ale změřit kvalitu vzorku a jak určit nejlepší vzorkovací metodu? Pro zhodnocení jak moc se vzorek podobá původnímu grafu je potřebné porovnat vlastnosti obou grafů. Budou se porovnávat distribuce uvedené v kapitole refsec:distribuce - distribuce stupňů vrcholů, distribuce velikostí komponent, hop-plot, hop-plot na největší komponentě, distribuce prvního levého singulárního vektoru, distribuce singulárních hodnot matice sousednosti a distribuce průměrného shlukovacího koeficientu pro všechny stupně vrcholů. Každá distribuce vlastnosti vzorku G_S je porovnávána s distribucí vlastnosti původního grafu G pomocí dvou výběrového Kolmogorovova-Smirnovova testu.

Kolmogorovův-Smirnovův test je metoda využívaná ve statistice, která umožňuje testovat, zda dvě náhodné proměnné pocházejí ze stejného rozdělení pravděpodobnosti, případně zda náhodná proměnná má předpokládané teoretické rozdělení [36]. Pro účely evaluace vzorkovacích metod je použita dvouvýběrová neparametrická varianta, která srovnává rozdělení dvou náhodných veličin. Z testu je použita pouze hodnota **D-value**, která slouží jako kritérium pro zamítnutí nulové hypotézy. Zde je použita pro určení podobnosti dvou distribucí. D-hodnota je definována podle vzorce 29.

$$D(P,Q) = \max_{x \in S}\{|P(x) - Q(x)|\} \tag{29}$$

P a Q jsou dvě kumulativní distribuční funkce a hodnota x je z množiny S, která představuje x-ové hodnoty distribuce nějaké vlastnosti vzorku. Hodnota x náleží do definičního oboru obou funkcí. D-hodnota zachycuje největší odchylku na ose y mezi kumulativními distribučními funkcemi P a Q. D-hodnota je použita jako měřítko podobnosti distribucí. Může nabývat hodnot $0 \leq D(P,Q) \leq 1$ a platí, že čím menší je D-hodnota pro danou distribuci, tím jsou si grafy v dané vlastnosti podobnější. $D(P,Q) = 0$ značí totožné distribuce $P = Q$.

Porovnávané distribuce nemají stejné měřítko, maximální hodnoty na ose x redukovaného grafu jsou mnohem menší než maximální hodnoty na ose x původního grafu. D-hodnota porovnává spíše tvar distribucí než jejich hodnoty. Proto je nutné data distribucí znormalizovat [2]. Nejprve se obě porovnávané distribuce převedou na kumulativní distribuce. Jestliže distribuce obsahuje na ose y záporné hodnoty, všechny hodnoty osy y se posunou do kladné části osy. Následně je osa x převedena na logaritmické měřítko a hodnoty se znormalizují do intervalu $0 \leq x \leq 1$ vydělením všech hodnot největší hodnotou v dané ose.

Pro účely zjištění D-hodnoty je implementovaná vlastní verze porovnání distribucí. Získaní vektorů, nad kterými se bude hledat maximální rozdíl je následující: první vektor je tvořen hodnotami na ose y relativní kumulativní distribuce P (distribuce nějaké vlastnosti vzorku). Druhý vektor je složen z y-ových hodnot distribuce Q (distribuce nějaké vlastnosti původního grafu) v x-ových bodech distribuce P. Následně je nalezen největší rozdíl mezi jednotlivými prvky vektorů.

5.3 Popis datových sad

5.3.1 Bezškálová a náhodná síť

Pro základní otestování chování jednotlivých metod byla vygenerovaná bezškálová síť pomocí Barabási–Albert modelu s parametry $n = 55000$, $n_0 = 100$ a $m = 3$. Výsledná síť má 55000 vrcholů a 164979 hran. Cílem bylo ověřit, zda vzorky vygenerované různými metodami zachovávají bezškálové vlastnosti.

Experiment se zaměřil i na náhodný graf vygenerovaný pomocí Erdős–Rényi modelu s parametry $n = 45000$ a $p = 0,0006$. Náhodný graf má 45000 vrcholů a 605989 hran. Cílem bylo zjistit, jak si vedou vzorkovací metody na náhodné sítí oproti reálným sítím.

5.3.2 Spoluautorská síť DBLP

Spoluautorská síť byla vytvořena z DBLP datasetu [37], který obsahuje základní bibliografické informace o publikacích se zaměřením na počítačové vědy. Data jsou volně dostupná ve formátu XML a v době psaní této práce (březen 2017) obsahovala 3688962 publikací s 1870930 autory. Pro potřeby vzorkování a experimentů by byla výsledná síť příliš velká, proto se ze zdrojového XML souboru extrahovaly publikace jen z roku 2011.

Výsledná neorientovaná síť obsahuje pouze největší komponentu vygenerované sítě. Vrcholy sítě představují autory a neorientovaná hrana mezi dvěma autory existuje v případě, že autoři jsou spoluautory alespoň jedné publikace. Více společných publikací dvou autorů je zanedbáno a váhy hran se rovnají jedné. Síť byla generována z publikací vydaných v roce 2011 a obsahuje 158632 vrcholů a 398521 hran.

5.3.3 Citační síť

Citační síť Arxiv HEP-PH (high energy physics phenomenology - publikace částicové fyziky) [38] je získána ze stránek s volným přístupem k více než milionu publikací z různých odvětví. Použitá citační síť pokrývá 34546 publikací a 421578 citací mezi těmito publikacemi. Jestliže publikace u cituje publikaci v, graf obsahuje orientovanou hranu z u do v. Orientace hran je pro účely experimentů zanedbána. Pokud publikace cituje jinou publikaci, která se nenachází v datasetu, informace o této citaci není v grafu zahrnuta. Data pocházejí z období od ledna 1993 do dubna 2003.

5.3.4 Elektrická síť

Elektrická síť na území západní části Spojených Států Amerických představuje typ technologické sítě. Obsahuje informace o elektrické síti na daném území. Hrana představuje elektrické vedení a vrchol představuje generátor, transformátor nebo rozvodnu. Síť byla původně použita v práci Wattse a Strogatze v roce 1998 [39]. Obsahuje 4941 vrcholů a 6594 neorientovaných hran.

5.3.5 Síť spolupráce (3-lambda)

Síť spolupráce byla vygenerována pomocí 3-lambda modelu, který byl představen v práci [40]. **3-lambda model** je pravděpodobností model, který pracuje s předpokladem, že jeden krok generace je jedna interakce zahrnující jak existující, tak i nově přidané vrcholy v daném kroku. Po každém kroku jsou vytvořeny hrany mezi všemi vrcholy, se kterými se v daném kroku pracuje. Některé hrany již mohou existovat i před interakcí. Vrcholy, které jsou zahrnuty v interakci mohou mít 4 různé role: klíčový vrchol interakce (proactive), sousedé klíčového vrcholu (neighbours), nově přidané hrany (newbies) a vrcholy, které nejsou sousedy klíčového vrcholu (new connections).

Každá interakce obsahuje vždy jen jeden klíčový vrchol a žádný nebo několik vrcholů ve třech jiných rolích. Počet vrcholů pro každý z těchto třech rolích je vybrán z Poissonova rozdělení s parametry λ_1 (pro výběr sousedů klíčového vrcholu), λ_2 (pro výběr nových vrcholů) a λ_3 (pro vrcholy, které nejsou sousedy klíčového vrcholu). Na počátku je kompletní graf o velikosti $(1+\lambda_1+\lambda_2+\lambda_3)$. V každém kroku je vybrán jeden klíčový vrchol a několik vrcholů ve zbývajících třech rolích podle zadaných parametrů λ. Mezi každým párem z těchto vrcholů je vytvořena hrana [40].

Pro účely experimentů byla vygenerovaná síť s parametry $\lambda_1 = 1,6$, $\lambda_2 = 0,35$, $\lambda_3 = 0,05$ a obsahuje 110000 vrcholů a 450528 hran.

5.3.6 Enron síť

Komunikační síť Enron [41] je sestavena z datasetu, který obsahuje kolem půl milionů emailů mezi 150 uživateli ve společnosti Enron. Vrcholy sítě představují emailové adresy. Neorientovaná

hrana mezi vrcholem i a j se vyskytuje v případě, že byl alespoň jednou odeslán email z adresy i na adresu j. Graf obsahuje 36692 vrcholů a 183831 hran.

5.4 Identifikace nejlepších vzorkovacích metod

Cílem tohoto experimentu je ověřit účinnost všech implementovaných metod na různých sítích a identifikovat nejlepší metody. Jako hlavní kritérium určující úspěšnost metody je to, jak se podobá tvar distribuce vlastnosti vzorku od tvaru distribuce vlastnosti původního grafu. Ukazatelem tohoto kritéria je D-hodnota představující maximální rozdíl mezi distribucemi (o postupu porovnávání pojednává kapitola 5.2). Pro vizuální porovnání distribucí je k dispozici série spojnicových grafů. Rovněž jsou u některých distribucí uvedené spojnicové grafy s relativní kumulativní četností, ze kterých se počítají D-hodnoty.

Velikost vzorků byla pro hlavní část experimentů stanovena na 15% původní velikosti sítě na základě pozorování z práce [2]. V kapitole 5.5 je zkoumáno, jaký vliv má velikost vzorku na přesnost různých metod. Pro každou metodu bylo vzorkování provedeno 5 krát. Výsledné D-hodnoty v tabulkách jsou průměrem těchto měření. Přiložené grafy zobrazují výsledky nejlepší iterace.

5.4.1 Citační síť

V tabulce 1 jsou uvedeny zprůměrované D-hodnoty pro každou distribuci vlastnosti z několika experimentů nad citační sítí. Pro každou metodu jsou D-hodnoty zprůměrovány do průměrné D-hodnoty (poslední sloupec), která udává celkovou úspěšnost metody. Tučně zvýrazněné hodnoty jsou nejlepší v rámci celého sloupce.

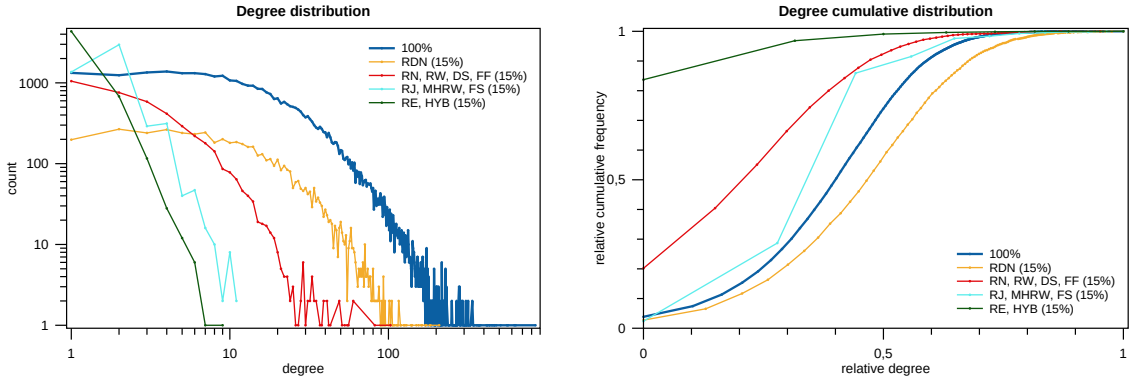
Tabulka 1: Průměrné D-hodnoty pro všechny metody a distribuce - **citační síť**.

	deg	wcc	clust	hops	hops on lc	sng-vec	sng-vals	average D
RN	0,353	**0,160**	0,147	0,078	0,078	0,235	**0,063**	0,159
RE	0,797	0,249	0,743	0,224	**0,058**	0,879	0,365	0,474
RDN	**0,165**	0,181	**0,133**	**0,020**	**0,020**	**0,040**	0,111	**0,096**
HYB	0,857	0,178	1,000	0,313	0,160	0,910	0,360	0,540
RW	0,555	1,000	0,159	0,077	0,077	0,232	0,153	0,322
RJ	0,291	0,290	0,179	0,235	0,235	0,526	0,251	0,287
MHRW	0,475	1,000	0,270	0,285	0,285	0,821	0,089	0,461
FF	0,525	1,000	**0,133**	0,127	0,127	0,175	0,109	0,314
DS	**0,134**	**0,161**	0,065	**0,049**	**0,049**	**0,043**	**0,036**	**0,077**
FS	0,629	0,620	0,357	0,214	0,214	0,562	0,185	0,397

V průměru si nejlépe vedla metoda Divided Stratums (DS) a metoda založená na výběru vrcholů Random degree node (RDN). Na obrázku 9(a) jsou znázorněny distribuce stupňů různých metod. Metody s podobným průběhem distribuce byly sloučeny tak, aby byla zachována přehlednost. I když D-hodnota pro distribuci stupňů u metody RDN je druhá nejlepší, z obrázku

je patrná nevýhoda, která plyne z chování RDN metody. Zvýhodňuje vrcholy s vyšším stupněm, takže konec křivky je vychýlen k původní distribuci stupňů více, než by měl.

U metod založených na procházení grafu (Random Walk, Divided Stratums, Forest Fire) a metodě Random Node (červená křivka) lze pozorovat, že nezvýhodňují vrcholy s velkým stupněm. Na druhou stranu je ve vzorku více vrcholů s nízkým stupněm, než by mělo být. Ideální distribuce stupňů by mohla vzniknout spojením metod RDN a například FF metody, kde by nebyly zvýhodněny vrcholy s nízkým ani s vysokým stupněm. Na obrázku 9(b) je znázorněna relativní kumulativní distribuce stupňů vrcholů, ze které se získává D-hodnota jako maximální rozdíl mezi dvěma kumulativními distribucemi.



(a) distribuce stupňů

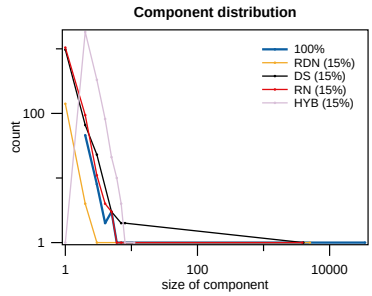
(b) kumulativní relativní distribuce stupňů

Obrázek 9: Porovnání distribucí stupňů původního grafu (modrá čára) a vzorků - **citační síť**.

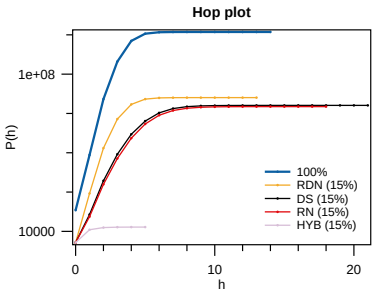
Nejlepší metoda pro citační síť Divided Stratums dobře zachovává hop-plot distribuce a velmi dobře zachovává distribuci prvního levého singulárního vektoru a distribuci vlastních čísel. Rovněž dobře zachovává distribuci stupňů a distribuci velikosti komponent, protože DS metoda zachovává strukturu původní sítě.

Ostatní metody založené na procházení grafů Forest Fire (FF), Metropolis-Hastings Random Walk (MHRW) a Random Walk (RW) nerespektují distribuci velikosti komponent a výsledný vzorek sestává pouze z jedné komponenty. Rovněž výsledky D-hodnoty distribuce vlastních čísel matice sousednosti nejsou u těchto metod nejlepší. Relativně dobře s ohledem na první levý singulární vektor a vlastní čísla si vedla metoda FF, která také nejlépe dokázala zachovat distribuci shlukovacího koeficientu.

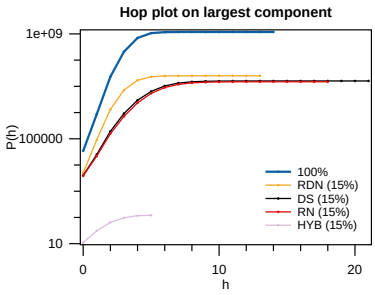
V sérii spojnicových grafů na obrázku 10 jsou zobrazeny distribuce všech vlastností citační sítě, ze kterých jsou zjišťované D-hodnoty v tabulce 1. Zobrazeny jsou distribuce z původního grafu (modrá čára), třech vzorků vytvořených nejlepšími metodami a jednoho vzorku vytvořeného nejhorší metodou. V případě citační sítě byla nejhorší metoda založena na výběru hran (HYB).



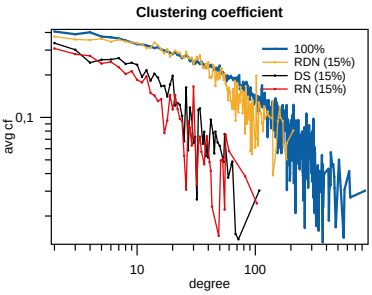
(a) distribuce velikost komponent



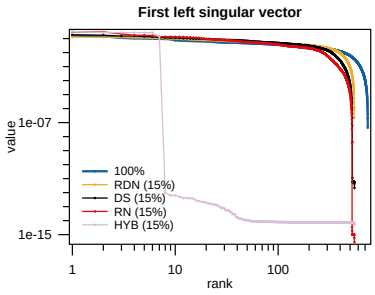
(b) Hop-plot



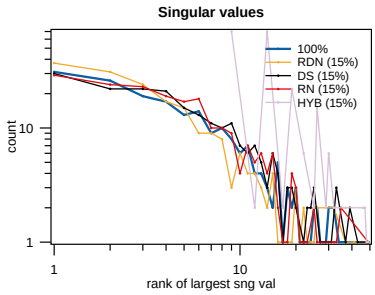
(c) Hop-plot na největší komponentě



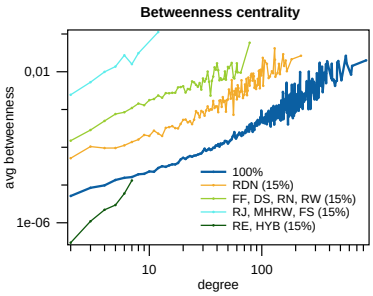
(d) distribuce shlukovacího koeficientu



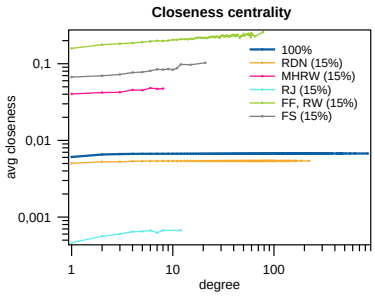
(e) První levý singulární vektor



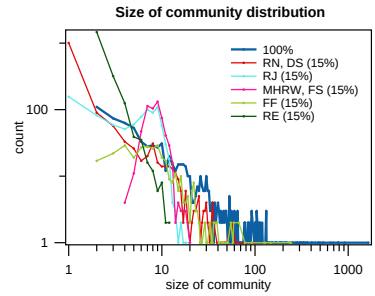
(f) Singulární hodnoty



(g) Betweenness centralita



(h) Closeness centralita



(i) Příslušnost ke komunitě

Obrázek 10: Všechny distribuce původního grafu a několika vzorků vytvořených různými metodami - **citační síť**.

Navíc jsou ještě uvedeny další dodatečné vlastnosti - betweenness a closeness centralita a distribuce velikosti komunit. Otázkou je, zda má u některých distribucí smysl měřit úspěšnost pomocí D-hodnoty. Při vizuálním prozkoumání grafů je vidět, že některé distribuce nemusí odpovídat původní distribuci a přitom mají relativně dobré D-hodnoty. Proto u posledních tří vlastností nebyly D-hodnoty zahrnuty v celkovém hodnocení metod.

- Z pohledu D-hodnot u **betweenness centrality** byla nejlepší metoda RDN, RJ a FF (D-hodnoty 0,146 a 0,175). Naopak nejhorší byla metoda RE (D-hodnota 0,381). Všeobecně nejhorší metody jsou založené na výběru hran, a to z důvodu rozpadu sítě na mnoho malých komponent. U betweenness centrality se to projevuje tak, že výsledná distribuce

u metod RE a HYB je „krátká“, protože maximální stupeň vrcholu u takto vytvořeného vzorku je menší než 10 a data tak nejsou dostatečná.

- V případě **closeness centrality** si vedly nejlépe metody založené na výběru vrcholů RDN a RN (D-hodnota 0,079 a 0,128). Z metod založených na procházení grafu si nejlépe vedlo DS (D-hodnota 0,186). Nejhorší metody byly opět RE a HYB (D-hodnota cca 0,3).
- **Komunitní strukturu** zachovává nejlépe metoda Forest Fire (D-hodnota 0,235). Nejhůře je na tom metoda MHRW (D-hodnota 0,822).

5.4.2 Bezškálová síť

Tabulka 2 obsahuje D-hodnoty pro bezškálovou síť vygenerovanou Barabási–Albert modelem. Podle průměrné D-hodnoty byla nejlepší metoda Forest Fire, která velmi dobře zachovává tvar hop-plot distribuce a obstojně zachovává tvar distribuce stupňů. V případě distribuce stupňů si lépe vedou metody MHRW, RJ a RDN.

Tabulka 2: Průměrné D-hodnoty pro všechny metody a distribuce - **bezškálová síť** vygenerovaná BA modelem.

	deg	wcc	clust	hops	hops on lc	sng-vec	sng-vals	average D
RN	0,204	1,000	0,624	**0,056**	**0,055**	0,372	0,142	0,351
RE	0,336	1,000	1,000	0,376	0,233	0,700	0,227	0,553
RDN	**0,065**	1,000	**0,109**	0,114	0,114	**0,026**	**0,046**	**0,211**
HYB	0,211	1,000	1,000	0,437	0,261	0,810	0,252	0,567
RW	0,315	**0,000**	0,470	**0,078**	**0,078**	0,298	0,190	**0,204**
RJ	0,118	1,000	0,439	**0,074**	**0,074**	0,432	**0,059**	0,314
MHRW	**0,074**	**0,000**	1,000	0,163	0,163	0,582	0,228	0,316
FF	0,136	**0,000**	0,439	**0,076**	**0,076**	0,345	0,182	**0,179**
DS	0,219	1,000	0,386	0,122	0,12	0,158	0,056	0,294
FS	0,310	1,000	0,841	**0,037**	**0,037**	0,322	0,109	0,379

Mezi další nejlepší metody podle průměrné D-hodnoty se řadí RW a RDN. RDN dobře zachovává první levý singulární vektor a singulární hodnoty. U singulárních hodnot si dobře vedla i metoda RJ. Nejhorší vzorky generovaly metody založené na výběru hran.

5.4.3 Náhodná síť

Pro ukázku toho, jak se vzorkovací metody chovají na sítích, které nemají vlastnosti reálných sítí, byly provedeny experimenty i na náhodné síti, která má Poissonovo rozdělení distribuce stupňů. Již při prvním pohledu na tabulku 3 lze pozorovat, že metody si v průměru nevedou tak dobře, jako u reálných sítí. Nejlépe si vedla metoda Forest Fire, ale například při pohledu na distribuci stupňů na obrázku 14(b) je vidět, že FF (spolu s RW) vůbec nezachovává zvonovitý tvar distribuce stupňů. Představené metody tak pro náhodné sítě při redukci na 15% původní velikosti nejsou vůbec vhodné.

Tabulka 3: Průměrné D-hodnoty pro všechny metody a distribuce - **náhodná síť**.

	deg	wcc	clust	hops	hops on lc	sng-vec	sng-vals	average D
RN	0,839	1,000	**0,284**	**0,033**	**0,033**	**0,153**	1,000	0,477
RE	0,891	1,000	1,000	0,467	0,325	0,986	1,000	0,810
RDN	0,821	1,000	**0,290**	**0,037**	**0,037**	**0,134**	1,000	0,474
HYB	**0,789**	1,000	1,000	0,466	0,330	0,992	1,000	0,797
RW	0,996	**0,000**	0,510	0,081	0,081	0,349	1,000	**0,431**
RJ	0,957	1,000	1,000	0,183	0,185	0,863	1,000	0,741
MHRW	0,991	**0,000**	1,000	0,184	0,184	0,753	1,000	0,587
FF	0,966	**0,000**	0,467	0,211	0,211	0,333	1,000	**0,455**
DS	0,968	1,000	**0,357**	**0,029**	**0,029**	0,298	1,000	0,526
FS	0,966	1,000	1,000	0,205	0,206	0,851	1,000	0,747

U distribuce shlukovacího koeficientu, hop-plot distribuce a distribuce prvního levého singulárního vektoru si metody založené na náhodném výběru vrcholů vedou lépe, než ostatní metody. Na obrázku 14(b) je pro ukázku uvedena distribuce stupňů ze vzorku o velikosti 40% původního grafu. Je vidět, jak se distribuce blíží k původní distribuci a při větších velikostech vzorku je distribuce více podobná původní distribuci.

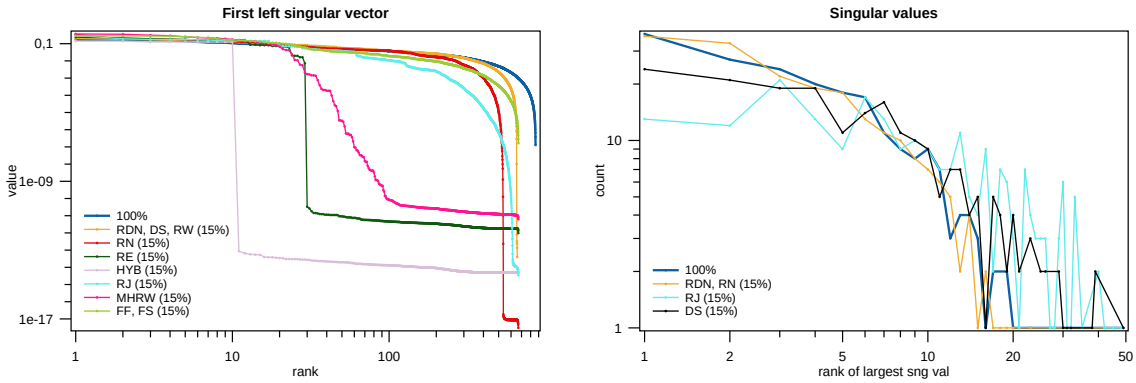
5.4.4 Síť spolupráce (3-lambda)

Pro síť spolupráce vygenerovanou modelem 3-lambda jsou výsledky D-hodnot pro vzorek o velikosti 15% původní velikosti uvedeny v tabulce 4. Nejlepší metody jsou založené na procházení grafu - Random Walk, Forest Fire a Divided Stratums. Opět je mezi nejlepšími metodami také metoda Random Degree Node, která dobře zachovává tvar distribuce stupňů. Nicméně i v tomto případě má výsledný vzorek vytvořený metodou RDN mnoho vrcholů s vysokým stupněm (viz obrázek 14(e)). Podobné D-hodnoty pro distribuci stupňů mají metody RJ, MHRW a DS, při kterých nevzniká nadbytečný počet vrcholů s vyšším stupněm. Podobně se distribuce stupňů chová i při metodě Forest Fire. U té je ale příliš mnoho vrcholů s nízkým stupněm.

Tabulka 4: Průměrné D-hodnoty pro všechny metody a distribuce - **síť spolupráce (3-lambda)**.

	deg	wcc	clust	hops	hops on lc	sng-vec	sng-vals	average D
RN	0,461	1,000	0,183	**0,042**	**0,042**	0,175	**0,062**	0,281
RE	0,797	1,000	0,180	0,231	0,065	0,744	0,216	0,462
RDN	**0,092**	1,000	**0,071**	0,068	0,068	**0,014**	**0,048**	**0,194**
HYB	0,853	1,000	1,000	0,372	0,192	0,867	0,298	0,654
RW	0,399	**0,000**	0,178	**0,034**	**0,034**	**0,062**	**0,092**	**0,114**
RJ	**0,103**	1,000	0,132	0,128	0,129	0,427	0,260	0,311
MHRW	**0,118**	**0,000**	0,275	0,213	0,213	0,756	0,226	0,257
FF	0,417	**0,000**	**0,091**	0,112	0,112	0,249	0,302	**0,183**
DS	**0,100**	1,000	**0,085**	**0,017**	**0,017**	**0,038**	0,164	**0,203**
FS	0,291	1,000	0,120	0,070	0,070	0,119	0,250	0,274

Na obrázku 11 jsou výsledky singulárního rozkladu matice sousednosti pro síť spolupráce. U prvního levého singulárního vektoru si nejlépe vedly metody RDN, DS a RW. Naopak nejhůře dopadly metody založené na výběru hran a MHRW. U distribuce singulárních hodnot si nejlépe vedly metody RN, RDN a RW (viz tabulka 4). Pro obě vlastnosti z obrázku 11 si nejlépe vedla metoda RW i RDN.



(a) první levý singulární vektor

(b) singulární hodnoty

Obrázek 11: Singulární rozklad matice sousednosti - **síť spolupráce (3 lambda)**.

Jelikož tyto vlastnosti dobře zachovávají distribuci prvního levého singulárního vektoru i distribuci singulárních hodnot (viz kapitola 5.1.7 a 5.1.8), dá se říct, že vzorky vygenerované těmito metodami zachovávají komunitní strukturu a topologii původní sítě.

5.4.5 Spoluautorská síť DBLP

U spoluautorské sítě DBLP jsou nejlepšími metodami, podobně jako u sítě spolupráce (3-lambda), Forest Fire a Random Walk. Průměrné D-hodnoty jsou uvedeny v tabulce 5. Distribuci stupňů vrcholů nejlépe zachovávají metody DS, MHRW, FS a RJ. V případě hop-plot distribuce si nejlépe vedou metody RDN, DS a RW.

Je zajímavé si povšimnout, jak se chová metoda DS, která si vede dobře nejen u hop-plot distribuce, ale i dobře zachovává distribuci vlastních čísel. Hůře si vede u distribuce prvního levého singulárního vektoru a nezachovává jednu velkou komponentu, ale rozděluje síť na několik komponent. Nejhůře si opět vede hybridní metoda založená na výběru hran, která rozloží síť na mnoho malých komponent.

Na obrázku 12 je distribuce velikostí komunit původní DBLP sítě a několika vzorků této sítě. Nejlépe zachovávala distribuci velikosti komunit metoda Forest Fire a Random Walk (D-hodnota 0,04). Dobře si vedla i metoda Frontier Sampling a MHRW. Naopak metody založené na výběru hran a vrcholů byly nejhorší.

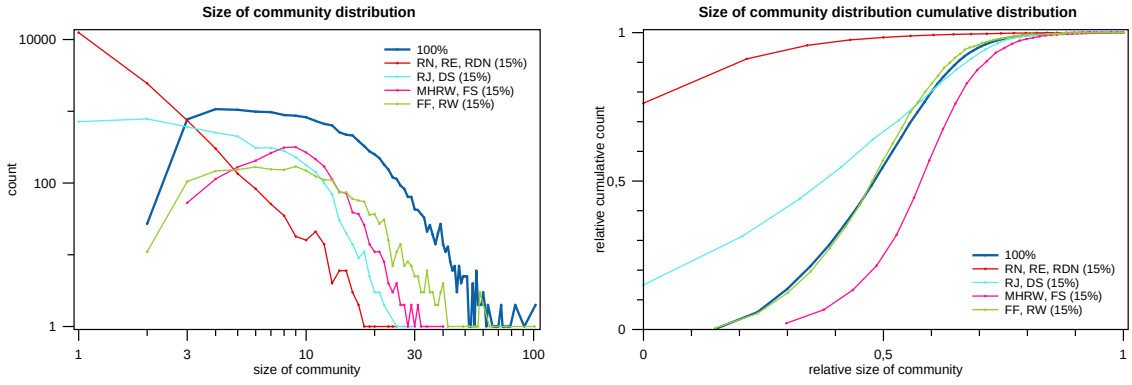
V tabulce 8 a 9 je přehled globálních vlastností původních sítí a několika vzorků nejlepších metod a jednou nejhorší metodou. U sítě DBLP je k dispozici dvojice hodnot (modularita)

Tabulka 5: Průměrné D-hodnoty pro všechny metody a distribuce - **spoluautorská síť DBLP**.

	deg	wcc	clust	hops	hops on lc	sng-vec	sng-vals	average D
RN	0,409	1,000	**0,124**	0,124	**0,035**	0,777	**0,114**	0,369
RE	0,759	1,000	**0,164**	0,261	0,064	0,540	0,272	0,437
RDN	0,205	1,000	0,205	**0,018**	**0,018**	**0,041**	0,146	0,233
HYB	0,802	1,000	0,630	0,395	0,278	0,875	0,338	0,617
RW	0,255	**0,000**	0,250	**0,032**	**0,032**	0,387	0,264	**0,174**
RJ	**0,158**	1,000	0,391	0,147	0,157	0,662	0,276	0,399
MHRW	**0,099**	**0,000**	0,313	0,185	0,185	0,505	0,140	0,204
FF	0,209	**0,000**	0,318	0,073	0,073	**0,072**	0,158	**0,129**
DS	**0,072**	1,000	0,220	**0,020**	**0,020**	0,402	**0,136**	0,267
FS	**0,121**	1,000	**0,188**	0,065	0,065	0,470	**0,070**	0,283

udávajících, jak dobře je síť rozdělená na komunity. Sítě s vysokou modularitou mají vysoce propojené vrcholy v rámci komunity, ale jen několik vazeb mezi různými komunitami. Příslušnost ke komunitám byla vypočítána pomocí algoritmů Infomap (Q_I) a Louvain (Q_L) (viz kapitola 4.3). V případě DBLP sítě všechny vzorky vytvořené pomocí nejlepších metod zachovávají modularitu sítě, přičemž nejlépe zachovává modularitu metoda Forest Fire. U bezškálové sítě (Barabási–Albert model) a náhodné sítě vytvářejí i nejlepší metody vzorky, které mají vyšší modularitu než původní sítě.

Z experimentů vyplývá, že metody založené na procházení grafu vytvářejí vzorky s mírně vyšší modularitou, než mají původní sítě. Může to být způsobeno tím, že při procházení grafu zůstane algoritmus spíše v husté komunitě a do další komunity se dostane s menší pravděpodobností.



(a) distribuce komunit

(b) kumulativní relativní funce distribuce komunit

Obrázek 12: Distribuce komunit podle Infomap - **síť DBLP**

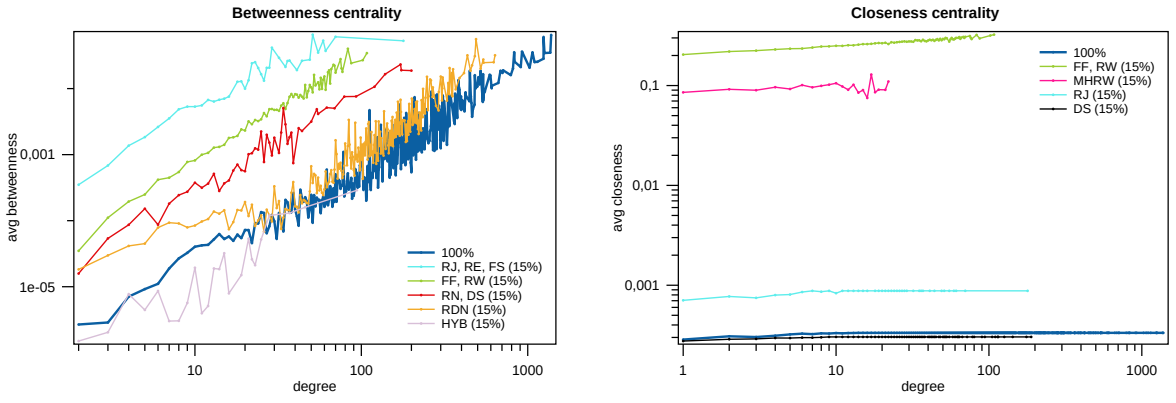
5.4.6 Enron síť

Experimenty nad Enron sítí ukázaly, že nejlepší vzorky generovala metoda Divided Stratums, která je založená na procházení grafu. Dobré průměrné D-hodnoty měly i vzorky generované metodami založenými na náhodném výběru vrcholů - RN a RDN (viz tabulka 6). Všechny zmíněné metody dobře zachovávají hop-plot distribuci, první levý singulární vektor i distribuci singulárních hodnot. RDN a DS rovněž dobře zachovává distribuci shlukovacího koeficientu. U distribuce stupňů vrcholů nejlépe vychází metody založené na procházení grafu - RW, FF, Frontier Sampling a nejlépe pak Divided Stratums.

Tabulka 6: Průměrné D-hodnoty pro všechny metody a distribuce - **enron síť**.

	deg	wcc	clust	hops	hops on lc	sng-vec	sng-vals	average D
RN	0,290	0,221	0,231	**0,021**	**0,022**	0,111	0,152	**0,150**
RE	0,405	**0,062**	0,241	0,142	0,142	0,325	0,484	0,257
RDN	0,364	0,233	**0,172**	0,044	0,044	**0,084**	**0,092**	**0,148**
HYB	0,561	0,170	0,817	0,109	0,106	0,482	0,256	0,357
RW	**0,151**	1,000	0,283	**0,023**	**0,023**	**0,084**	**0,120**	0,240
RJ	0,274	**0,106**	0,584	0,089	0,089	0,280	0,224	0,235
MHRW	0,198	1,000	0,274	0,173	0,173	0,647	0,267	0,390
FF	**0,169**	1,000	**0,153**	0,081	0,081	**0,083**	0,248	0,259
DS	**0,081**	0,261	**0,112**	**0,011**	**0,011**	**0,059**	**0,080**	**0,089**
FS	**0,106**	0,444	0,390	0,062	0,062	**0,083**	0,429	0,225

Na obrázku 13 je zobrazena distribuce betweenness a closeness centrality Enron sítě. Nejlépe zachovává distribuci closeness centrality metoda Divided Stratums (D-hodnota 0,094). Co je více u těchto centralit důležitější než D-hodnota, je samotná hodnota centralit. Metoda DS zachovávala průměrné hodnoty pro jednotlivé stupně vrcholů. Metody jako FF, RW, RJ a MHRW vytvářely vzorky, které měly řádově vyšší průměrné closeness centrality pro dané stupně vrcholů.



(a) betweenness centrality

(b) closeness centrality

Obrázek 13: Distribuce centralit - **Enron síť**.

U betweenness centrality byla nejlepší metoda RDN (D-hodnota 0,101). Překvapivě dobře si vedla i metoda RE (D-hodnota 0,117), důležitější jsou ovšem samotné hodnoty centralit. Zatímco metoda RDN vytvářela vzorky s jen o něco vyšší průměrnou betweenness centralitou pro daný stupeň, vzorky ostatních metod měly vyšší průměrnou betweenness centrality pro daný stupeň vrcholu.

5.4.7 Elektrická síť

Průměrné D-hodnoty pro poslední testovanou síť - elektrickou síť jsou uvedené v tabulce 7. Opět jsou nejlepší metody založené na procházení grafu - Metropolis-Hastings Random Walk, RW a FF, které měly D-hodnotu nejlepší téměř ve všech distribucích. Tyto metody zachovávaly dobře distribuci stupňů vrcholů.

Tabulka 7: Průměrné D-hodnoty pro všechny metody a distribuce - **technologická (elektrická) síť**.

	deg	wcc	clust	hops	hops on lc	sng-vec	sng-vals	average D
RN	0,430	1,000	0,840	0,517	0,376	0,710	0,742	0,659
RE	0,733	1,000	1,000	0,529	0,417	0,707	0,428	0,688
RDN	0,236	1,000	0,193	0,391	0,263	0,489	0,365	0,419
HYB	0,681	1,000	1,000	0,555	0,459	0,729	0,512	0,705
RW	**0,070**	**0,000**	0,165	**0,090**	**0,090**	0,216	**0,311**	**0,134**
RJ	0,259	1,000	**0,138**	0,300	0,197	0,470	0,432	0,399
MHRW	**0,070**	**0,000**	**0,140**	**0,069**	**0,069**	**0,101**	0,401	**0,122**
FF	**0,075**	**0,000**	**0,133**	**0,049**	**0,049**	0,279	**0,332**	**0,131**
DS $p = 0,5$	**0,058**	1,000	0,182	0,380	0,276	0,603	0,518	0,431
DS $p = 0,9$	0,210	1,000	**0,120**	0,216	0,134	0,219	0,284	0,312
FS	0,508	1,000	0,826	0,435	0,315	0,616	0,786	0,641

Distribuci stupňů vrcholů zachovávala dobře i metoda Divided Stratums, ale jen při nižším parametru p. Ostatní metody u této sítě si nevedly tak dobře. Například metoda RDN, která si jinak vedla dobře, zachovává relativně dobře jen distribuci shlukovacího koeficientu.

5.4.8 Asortativita

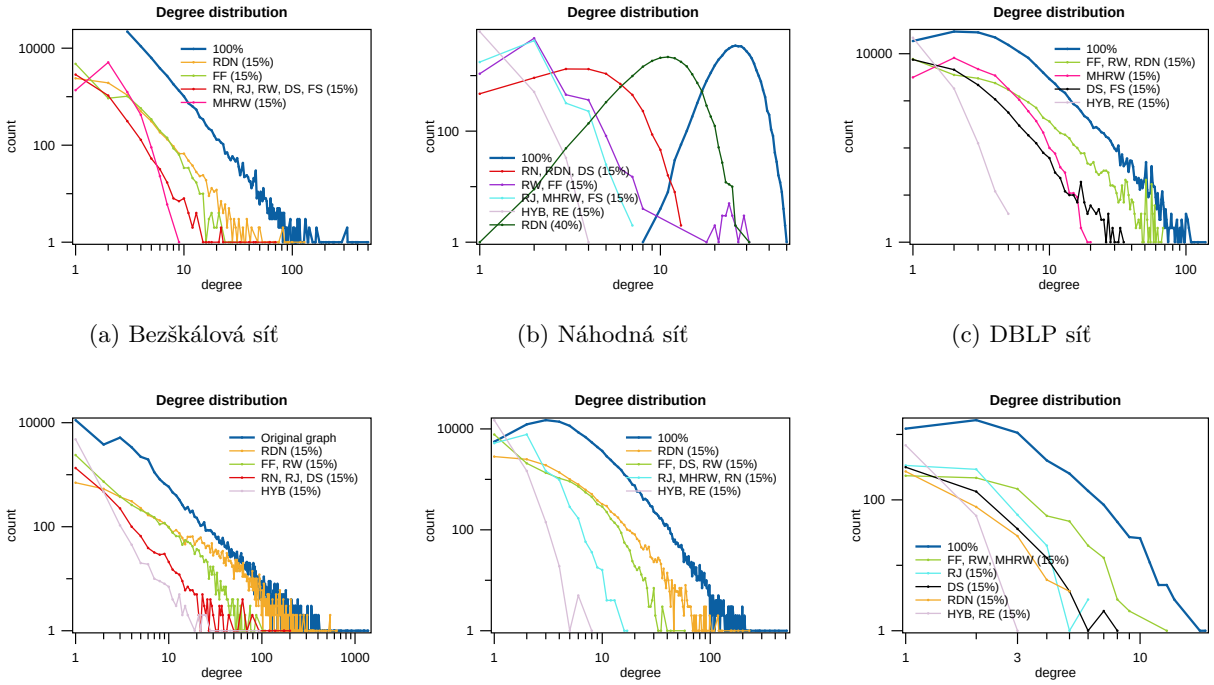
Asortativita je souhrnná statistika propojenosti uzlů. Nižší hodnoty indikují, že převládají hrany spojující vrcholy s rozdílnými stupni. Asortativita může nabývat hodnot v intervalu $\langle -1, 1 \rangle$. Kladné číslo indikuje korelaci mezi vrcholy s podobným stupněm, zatímco záporná hodnota indikuje vztahy mezi vrcholy s odlišným stupněm vrcholu. Asortativita také poskytuje informaci o topologické struktuře sítě [42]. Bylo zjištěno, že sociální sítě nebo sítě spolupráce mají kladnou asortativitu, naopak biologické nebo technické sítě mají spíše zápornou asorativitu.

Bezškálová síť vygenerovaná Barabási–Albert modelem a náhodná síť mají asorativitu kolem 0 [43]. Odpovídá to i změřeným hodnotám v tabulce 8 a 9, ve které má bezškálová i náhodná síť asoratitvitu blízkou 0 a sítě spolupráce kladnou hodnotu. Enron síť má zápornou hodnotu. V

případě DBLP sítě nejlépe asorativitu (0,451) zachovává metoda MHRW (0,458), dobře si vede i metoda FF (0,668). Naopak HYB metoda tvoří vzorek s nulovou asortativitou. U sítě spolupráce (3-lambda) dobře zachovávají asortativitu metody RW a RDN (hodnota 0,145). Metoda FF vytvořila vzorek se zápornou asortativitou.

5.4.9 Porovnání distribucí stupňů

Na obrázku 14 jsou distribuce stupňů pro všechny testované grafy. U bezškálové sítě (a), enron sítě (d) a sítě spolupráce DBLP (e) je jasně vidět, že metoda Random Degree Node zvýhodňuje vrcholy s vysokým stupněm. Ve výsledném vzorku je tak více vrcholů s vysokým stupněm, než by mělo být. Konec křivky se více přibližuje distribuci původního grafu. Naproti tomu, metoda Forest Fire a další metody založené na procházení grafu tímto neduhem netrpí. U sítě DBLP byly distribuce stupňů vrcholů pro metody FF i RDN velmi podobné. U této sítě si z hlediska distribuce stupňů vedla nejlépe metoda DS a MHRW.



(a) Bezškálová síť (b) Náhodná síť (c) DBLP síť

(d) Enron síť (e) síť spolupráce (3-lambda) (f) Elektrická síť

Obrázek 14: Distribuce stupňů jednotlivých grafů a jejich vzorků (15%)

U náhodné sítě (b) je názorně vidět, jak metody založené na výběru hran a na procházení grafu nerespektují Poissonovo rozdělení. Nejblíž k původnímu tvaru distribuce mají metody RN, RDN a DS. Ze všech grafů vyplývá, že distribuce u metod RE a HYB mají „strmý pád“ - mnoho vrcholů s nízkým stupněm a jen málo s vysokým stupněm. Výsledný vzorek obsahuje mnoho malých komponent a vzorky tak nerespektují původní distribuci velikosti komponent.

Tabulka 8: Souhrnné globální vlastnosti všech testovaných původních sítí a jejich vzorky pořízené nejlepšími a jednou nejhorší metodou, první část. $|V|$ - *počet vrcholů,* $|E|$ - *počet hran,* D - *hustota,* $\langle k \rangle$ - *průměrný stupeň,* d - *průměr,* l - *průměrná délka cesty,* $\langle C \rangle$ - *průměrný shlukovací koeficient,* A - *asortativita,* Q_I - *modularita (Infomap),* Q_L - *modularita (Louvain).*

	$\|V\|$	$\|E\|$	D	$\langle k \rangle$	d	l	# of comp	largest comp	$\langle C \rangle$	A	Q_I	Q_L
Citační síť(100%)	**34546**	**420921**	**7,10E-04**	**24,37**	**14**	**4,33**	**61**	**34401**	**0,285**	**-0,006**	**0,62**	**0,72**
1. DS (15%)	5221	19727	1,45E-03	7,56	15	5,16	241	4949	0,244	0,011	0,66	0,75
2. RDN (15%)	5182	42232	3,15E-03	16,3	14	4,12	167	5000	0,285	0,039	0,64	0,71
3. RN (15%)	5128	9930	7,40E-04	3,83	17	6,28	1101	3940	0,157	0,012	0,68	0,78
4. RJ (15%)	5182	4953	3,70E-04	1,91	84	27,79	436	3705	0,010	-0,019	0,88	0,96
10. HYB (15%)	5182	2894	2,20E-04	1,12	7	1,30	2288	10	0,000	-0,029	1,00	1,00
Bezškálová síť (100%)	**55000**	**164979**	**1,10E-04**	**5,99**	**8**	**4,87**	**1**	**55000**	**0,001**	**-0,027**	**0,31**	**0,38**
1. FF (15%)	8250	9495	2,80E-04	2,3	20	9,82	1	8250	0,000	-0,175	0,77	0,86
2. RW (15%)	8250	9544	2,80E-04	2,31	33	9,17	1	8250	0,000	-0,003	0,76	0,85
3. RDN (15%)	8250	11755	3,50E-04	2,85	17	5,23	1305	6850	0,004	-0,043	0,52	0,61
4. DS (15%)	8251	10330	3,00E-04	2,50	16	5,75	494	7689	0,002	-0,116	0,67	0,75
10. HYB (15%)	8250	4665	1,40E-04	1,13	7	1,43	3585	18	0,000	-0,032	1,00	1,00
3-lambda (100%)	**110000**	**450528**	**7,00E-05**	**8,19**	**14**	**5,58**	**1**	**110000**	**0,668**	**0,145**	**0,67**	**0,73**
1. RW (15%)	16500	25949	1,90E-04	3,15	32	8,03	1	16500	0,160	0,160	0,76	0,86
2. FF (15%)	16500	28328	2,10E-04	3,43	17	7,92	1	16500	0,189	-0,082	0,80	0,89
3. RDN (15%)	16500	46846	3,40E-04	5,68	14	4,98	2274	13760	0,372	0,180	0,56	0,64
4. DS (15%)	16499	31066	2,30E-04	3,77	18	6,61	1308	14613	0,460	0,039	0,79	0,88
10. HYB (15%)	16501	9153	7,00E-05	1,11	5	1,30	7354	12	0,001	0,019	1,00	1,00
Náhodná síť (100%)	**45000**	**605989**	**6,00E-04**	**26,94**	**5**	**3,62**	**1**	**45000**	**0,001**	**-0,001**	**0,09**	**0,15**
1. RW (15%)	6750	7291	3,20E-04	2,16	48	10,89	1	6750	0,000	0,102	0,81	0,91
2. FF (15%)	6750	7257	3,20E-04	2,15	15	9,47	1	6750	0,000	-0,389	0,82	0,91
3. RDN (15%)	6750	14559	6,40E-04	4,31	12	6,17	92	6656	0,000	-0,004	0,43	0,49
4. RN (15%)	6750	13707	6,00E-04	4,06	13	6,41	107	6635	0,001	0,005	0,46	0,52
10. RE (15%)	6751	3654	1,60E-04	1,08	4	1,18	3097	6	0,000	-0,025	1,00	1,00

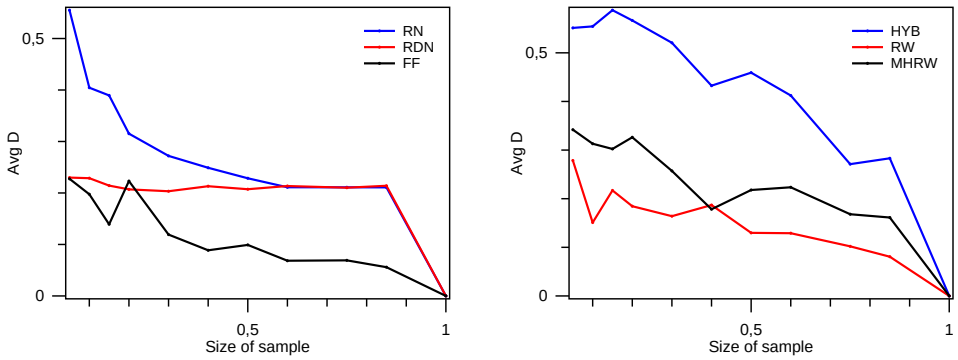
Tabulka 9: Souhrnné globální vlastnosti všech testovaných původních sítí a jejich vzorky pořízené nejlepšími a jednou nejhorší metodou, druhá část. $|V|$ - *počet vrcholů,* $|E|$ - *počet hran,* D - *hustota,* $\langle k \rangle$ - *průměrný stupeň,* d - *průměr,* l - *průměrná délka cesty,* $\langle C \rangle$ - *průměrný shlukovací koeficient,* A - *asortativita,* Q_I - *modularita (Infomap),* Q_L - *modularita (Louvain).*

	$\|V\|$	$\|E\|$	D	$\langle k \rangle$	d	l	# of comp	largest comp	$\langle C \rangle$	A	Q_I	Q_L
DBLP síť (100%)	**158632**	**398521**	**3,00E-05**	**5,02**	**37**	**10,18**	**1**	**158632**	**0,562**	**0,451**	**0,85**	**0,93**
1. FF (15%)	23795	51328	1,80E-04	4,31	26	10,39	1	23795	0,313	0,668	0,86	0,93
2. RW (15%)	23795	37731	1,30E-04	3,17	35	8,48	1	23795	0,234	0,129	0,78	0,88
3. MHRW (15%)	23795	37523	1,30E-04	3,15	67	26,59	1	23795	0,344	0,459	0,90	0,97
4. RDN (15%)	23795	36078	1,30E-04	3,03	28	8,77	9294	8732	0,278	0,848	0,90	0,94
10. HYB (15%)	23796	12962	5,00E-05	1,09	7	1,20	10849	8	0,002	-0,007	1,00	1,00
Enron síť (100%)	**36692**	**183831**	**2,70E-04**	**10,02**	**13**	**4,03**	**1065**	**33696**	**0,497**	**-0,111**	**0,53**	**0,59**
1. DS (15%)	6199	16074	8,40E-04	5,19	11	3,89	1478	4621	0,298	-0,140	0,50	0,56
2. RDN (15%)	5504	69614	4,60E-03	25,3	11	3,24	346	5083	0,433	-0,080	0,44	0,46
3. RN (15%)	5504	4462	2,90E-04	1,62	11	4,01	3279	1817	0,107	-0,136	0,62	0,68
4. FS (15%)	5504	9137	6,00E-04	3,32	22	6,21	74	5155	0,075	-0,110	0,60	0,70
10. MHRW (15%)	5504	8321	5,50E-04	3,02	34	11,13	1	5504	0,193	0,071	0,78	0,87
Elektrická síť (100%)	**4941**	**6594**	**5,40E-04**	**2,67**	**46**	**18,99**	**1**	**4941**	**0,080**	**0,003**	**0,82**	**0,94**
1. MHRW (15%)	741	852	3,11E-03	2,3	100	38,27	1	741	0,041	-0,052	0,82	0,90
2. FF (15%)	741	919	3,55E-03	2,48	35	16,19	1	741	0,101	-0,053	0,83	0,90
3. RW (15%)	741	903	3,29E-03	2,44	29	11,83	1	741	0,027	0,002	0,78	0,86
4. DS (15%, $p = 0,9$)	736	683	2,53E-03	1,86	25	7,25	121	88	0,055	-0,001	0,90	0,95
10. HYB (15%)	741	400	1,46E-03	1,08	3	1,15	341	4	0,000	-0,056	1,00	1,00

5.5 Identifikace optimální velikosti vzorku

Tato kapitola zkoumá, jak se chovají různé metody vzorkování grafů při různých velikostech vzorků. Cílem je identifikovat minimální velikost vzorku, při které vytváří daná metoda dostatečně dobrý vzorek. Kvalita vzorku je opět určená průměrnou D-hodnotou ze všech distribucí. V každém testu bylo vytvořeno celkem 10 vzorků různých velikostí - 85%, 75%, 60%, 50%, 40%, 30%, 20%, 15%, 10% a 5% původní velikosti grafu. Experiment byl proveden nad bezškálovou sítí vygenerovanou Barabási-Albert modelem a elektrickou sítí.

Na dvojici obrázků 15 je porovnání různých metod, které generují různě velké vzorky z bezškálového grafu. Na prvním obrázku jsou metody RN, RDN a FF. Nejlépe napříč všemi velikostmi si vedla metoda FF. Při nejnižších velikostech (5%-20%) průměrná D-hodnota v případě FF stoupla a vyrovnala se D-hodnotě metody RDN. Minimální velikost vzorku pro metodu FF je tedy v případě bezškálové sítě cca 25% (d-hodnota cca 0,12). Metody RN a RDN se při velikosti větší než 55% chovají stejně. Při nižších velikostech již metoda RN nezvládá generovat tak dobré vzorky jako metoda RDN. Metoda RDN navíc zachovává přibližně stejnou průměrnou D-hodnotu napříč všemi velikostmi vzorku (D-hodnota kolem 0,20).



(a) porovnání metod založených na výběru vrcholů (RN, RDN) s FF metodou

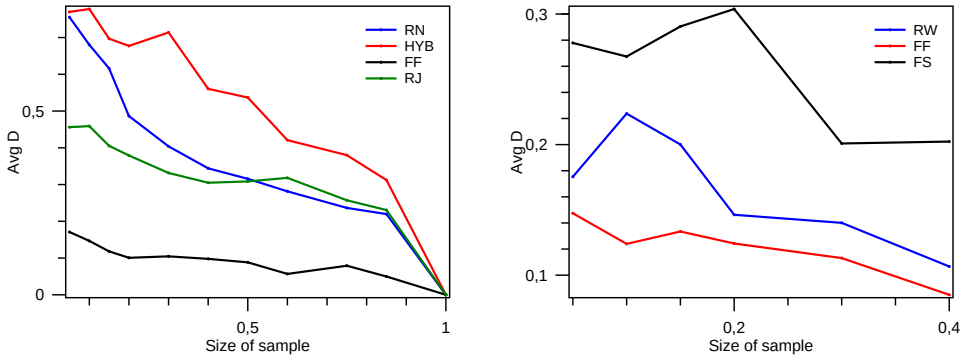
(b) porovnání metod založených na procházení grafu (RW, MHRW) s HYB metodou.

Obrázek 15: Průměrná D-hodnota pro různé velikosti vzorků a různé metody - **bezškálový graf**.

Na obrázku 15(b) je porovnávána metoda RW, MHRW a HYB. Metoda HYB založená na výběru hran generuje vzorky s průměrnou D-hodnotou horší než u ostatních metod již od 85% velikosti původní sítě. Při nižších velikostech se D-hodnota výrazně zhoršuje. Metoda MHRW generovala nejlepší vzorky při 40% původní velikosti grafu (D-hodnota 0,178). Při větší velikosti ($> 40\%$) se metoda MHRW zhoršovala více než metoda RW. Metoda RW si zachovávala relativně dobrou průměrnou D-hodnotu (0,15) i při velikosti 10% původní sítě.

Na obrázku 16(a) jsou výsledky experimentů nad grafem elektrické sítě. Nejhůře napříč všemi velikostmi si vedla opět metoda HYB. Metody RN a RJ generují vzorky s přibližně stejnou průměrnou D-hodnotou do cca 50% velikosti původního grafu. Při nižších velikostech si vede

metoda RN hůře než RJ a postupně se s menšími velikostmi zhoršuje až na úroveň HYB metody (D-hodnota 0,75).



(a) RN, HYB, FF a RJ metoda - elektrická síť (b) RW, FF a FS metoda - DBLP síť

Obrázek 16: Průměrná D-hodnota pro různé velikosti vzorků - **elektrická síť** a **DBLP síť**

Nejlépe vychází opět metoda FF, která i při velikosti 20% původního grafu generuje vzorky s průměrnou D-hodnotou 0,1 (při 10% - 0,14, při 5% - 0,17). V práci [2] je v případě FF metody stanovena optimální velikost 15% původního grafu. U těchto experimentů byla stanovena optimální velikost na 20 až 25%. Je možné, že se optimální velikost může ještě snížit při jiných typech sítí s větší velikostí. Pro ukázku byla ještě otestována spoluautorská síť DBLP (do 40% původní velikosti sítě - obrázek 16(b)). Při velikosti vzorku kolem 20% původní sítě si ještě dobře vede metoda FF a RW. Metoda FS vytváří optimální vzorky o velikosti 30% původní sítě.

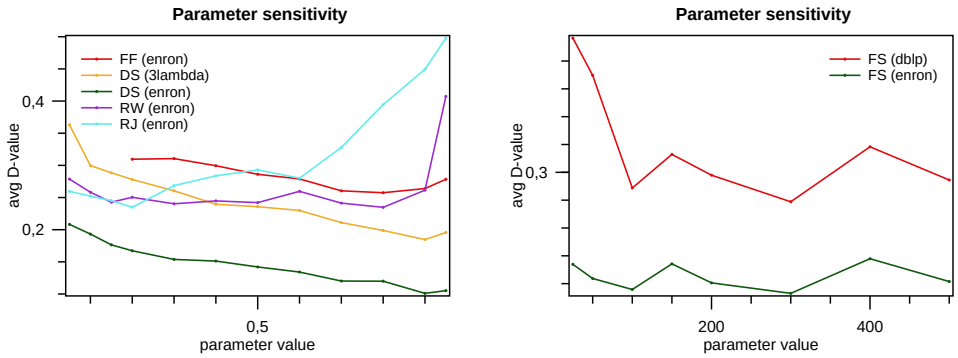
5.6 Experimenty s parametry vzorkovacích metod

Tato kapitola se zaměřuje na testování metod založených na procházení grafu s různými hodnotami nastavitelných parametrů. Zkoumá se vliv hodnoty parametru na kvalitu vygenerovaného vzorku. Jako ukazatel kvality vzorku je opět použita průměrná D-hodnota. Experiment byl proveden nad sítí Enron s metodami Random Walk, Random Jump, Divided Stratums a Forest Fire. Další testovaná metoda byla Frontier Sampling, která vygenerovala vzorek z Enron sítě a DBLP sítě. Pracovalo se se vzorkem o velikosti 15% původního grafu.

U metody Forest Fire se pracovalo s parametrem p_f, který představuje pravděpodobnost „zapálení“ sousedních vrcholů. U metody Random Walk a Random Jump je to parametr c, který udává pravděpodobnost restartu procházky v každém kroku (RW) nebo pravděpodobnost skoku na jiný vrchol sítě v každém kroku (RJ). Metoda Divided Stratums má parametr k, udávající pravděpodobnost výběru vrcholů z množiny vrcholů V_{i_adj}, jenž obsahuje vrcholy propojené hranou s vrcholy, které již patří do vzorku. Všechny tyto parametry nabývají hodnot od 0 do 1.

5.6.1 Enron síť, síť spolupráce (DS metoda) a porovnání FS metody nad Enron a DBLP síti)

Na obrázku 17(a) jsou výsledky metod s různými hodnotami parametrů. Pro Forest Fire byla D-hodnota nejlepší při vyšších hodnotách parametru $(0,7 < p_f < 0,9)$. Random Jump metoda generovala nejlepší vzorky s nižší hodnotou parametru $(c = 0,2)$. Největší vliv na D-hodnotu u Enron sítě měl parametr u metody Random Jump, protože při vyšších hodnotách c parametru si vedla nejhůře a rozdíl oproti nejlepší a nejhorší D-hodnotě byl největší. V případě metody Random Walk ani tak nezáleželo na hodnotě parametru. Dobře si v případě Enron sítě vedla v širokém rozmezí $0,15 < c < 0,9$. Metoda Divided Stratums si nejlépe vedla při vyšších hodnotách parametru $(k = 0,9)$. Pro ověření je ještě v grafu uvedena metoda Divided Stratums pro síť spolupráce vygenerovanou 3-lambda modelem. Výsledky jsou podobné, jako u Enron sítě.



(a) **síť Enron** a DS pro síť **3-lambda**

(b) Porovnání FS metody na síti **Enron** a **DBLP** (počet počátečních vrcholů n_0 v závislosti na parametru p viz vzorec 30)

Obrázek 17: Průměrná D-hodnota v závislosti na hodnotě parametru.

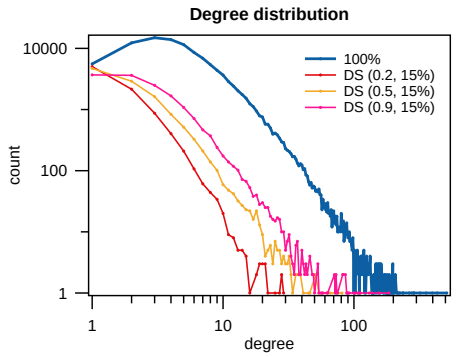
Na obrázku 17(b) je porovnání úspěšnosti metody Frontier Sampling v závislosti na parametru p nad sítí Enron a DBLP. Parametr p určuje počet počátečních vrcholů n_0 podle následujícího vzorce 30.

$$n_0 = \frac{|V|}{p} \tag{30}$$

Čím vyšší byl p parametr, tím nižší byl počáteční počet vrcholů. U sítě Enron nelze jednoznačně určit správnou hodnotu parametru. Nejlepší byla D-hodnota při $p = 100$ a $p = 300$ (D-hodnota cca 0,21). Nicméně rozptyl D-hodnot pro FS byl minimální. U větší sítě DBLP již lze pozorovat, že při nízkých hodnotách parametru se průměrná D-hodnota razantně zhoršuje. Nejnižší D-hodnoty byly opět u hodnot 100 a 300. V případě metody Frontier Sampling tak záleží na velikosti původní sítě, kdy při rozsáhlejší síti jsou rozdíly způsobené rozdílnými hodnotami parametru větší.

5.6.2 Metoda DS nad sítí spolupráce s různou hodnotou parametru

Obrázek 18 ukazuje, jak se chovají distribuce stupňů vzorků vygenerovaných metodou Divided Stratums ze sítě spolupráce (3-lambda) v závislosti na různých hodnotách parametru p. Se zvyšující se hodnotou parametru se křivka přibližuje k původní distribuci. Pro $p = 0,2$ byla D-hodnota 0,405, pro $p = 0,5$ - 0,314 a pro $p = 0,9$ - 0,114. U technologické sítě (elektrické) si s ohledem na průměrnou D-hodnotu lépe vede metoda Divided Stratums s parametrem $p = 0,9$. Nicméně distribuci stupňů lépe zachovává s parametrem $p = 0,5$ (tabulka 7).



Obrázek 18: Porovnání distribuce stupňů pro metodu DS s různou hodnotou parametru p - **síť spolupráce (3-lambda)**.

5.7 Topologická struktura sítě

V příloze A je na sérii obrázků znázorněna elektrická síť a vzorky této sítě, vytvořené všemi implementovanými metodami. Velikost vzorku je 15% původní velikosti sítě. Obrázky dávají přehled o různých rozvrženích, které jednotlivé metody vytvářejí. Lze tak porovnat jednotlivé topologické struktury sítí, které metody vytvářejí.

Metody založené na náhodném výběru vrcholů nebo hran (RN, RDN, RE a HYB) vytvářejí vzorky s mnoha izolovanými vrcholy a nezachovávají původní topologickou strukturu. Metody založené na procházení grafu již vypadají lépe. U RW a MHRW lze pozorovat, jak se tvořila cesta v původní síti, algoritmus se tedy nemusí dostat do všech částí sítě. Podobně se chová i algoritmus FF metody, který se nedostane do odlehlejších částí původní sítě. U vzorků z metod RJ a FS to již vypadá, že zachovávají původní strukturu sítě, protože algoritmus v případě FS provádí několik náhodných procházek. V případě RJ skáče algoritmus na jiné vrcholy v sítí.

Nejlépe s ohledem na strukturu sítě vypadá vzorek vytvořený pomocí metody Divided Stratums. Hlavní souvislá komponenta je rozprostřena napříč celým původním grafem, přitom vzorek obsahuje i pár oddělených komponent z odlehlejších částí grafu. Podobně se chová i metoda FS. Zde ale není graf tak propojen, jako v případě DS.

V tabulce 10 je uvedena asortativita a modularita u vybraných vzorků. Asortativita poskytuje informaci o topologické struktuře sítě. Modularita měří komunitní strukturu sítě. Metoda

DS dobře zachovávala tyto vlastnosti, pouze u sítě spolupráce vygenerované 3-lambda modelem byly větší odchylky. V některých případech si dobře vedla i metoda RW, která měla podobné hodnoty, jako metoda DS.

Tabulka 10: Asortativita a modularita u vybraných vzorků.

	A	Q_I	Q_L
Citační síť (100%)	**-0,006**	**0,62**	**0,72**
DS	0,011	**0,66**	**0,75**
RW	0,181	0,69	0,78
RJ	**-0,019**	0,88	0,96
FS	-0,028	0,81	0,91
Síť spolupráce (100%)	**0,145**	**0,67**	**0,73**
DS	0,039	0,79	0,88
RW	**0,16**	**0,76**	**0,86**
RJ	0,036	0,91	0,97
FS	0,036	0,83	0,91
DBLP síť (100%)	**0,451**	**0,85**	**0,93**
DS	**0,466**	**0,91**	**0,97**
RW	0,129	0,78	0,88
RJ	0,07	0,97	0,99
FS	0,489	0,9	**0,97**
Enron síť (100%)	**-0,111**	**0,53**	**0,59**
DS	-0,140	0,50	0,56
RW	-0,041	**0,51**	**0,59**
RJ	-0,088	0,74	0,82
FS	**-0,110**	0,60	0,70
Elektrická síť (100%)	**0,003**	**0,82**	**0,94**
DS	-0,001	0,90	**0,95**
RW	**0,002**	**0,78**	0,86
RJ	-0,090	0,95	0,98
FS	0,059	0,99	0,99

6 Závěr

Generování menších vzorků, které odpovídají původní síti, je důležitý nástroj pro analýzu rozsáhlých sítí. Cílem této práce bylo seznámení s problematikou vzorkování komplexních sítí, implementace některých algoritmů pro vzorkování a experimenty nad vybraními sítěmi. Cílem experimentů bylo určení metod vytvářející vzorky, které nejlépe zachovávají vlastnosti původní sítě. V rámci práce byly popsány vlastnosti komplexních sítí, které se při vzorkování sledují. Práce rovněž obsahuje uvedení do problematiky vzorkování a přehledně popsané metody vzorkování. Podařilo se implementovat nástroj, který umožňuje vytváření vzorků pomocí deseti implementovaných algoritmů. Nástroj umožňuje zjistit globální vlastnosti grafu a distribuce vlastností, které se porovnávají. Distribuce lze přímo v nástroji porovnávat jak statisticky, tak i vizuálně. Experimentální část se věnuje porovnání jednotlivých implementovaných metod a obsahuje naměřené D-hodnoty a grafy pro vizuální porovnání distribucí. Experimenty byly provedeny nad celkem 6 reálnými sítěmi a náhodným grafem. Závěry z experimentální části jsou následující:

- Nejlepší vzorky generovaly metody založené na procházení grafu. Nejnižší průměrnou D-hodnotu při 15% velikosti původní sítě měly vzorky vygenerované metodou **Divided Stratums**, **Forest Fire** a **Random Walk**. Z pohledu průměrných D-hodnot si dobře vedla i metoda Random Degree Node, jen generovala vzorky s příliš mnoha vrcholy s vysokým stupněm. Nejhorší vzorky generovaly metody založené na výběru hran, zejména hybridní metoda. Zkoumané metody generují dobré vzorky z reálných sítích. U náhodné sítě metody nefungují.

- Optimální velikost vzorku byla pomocí měření nad grafy o velikosti 55000 a 4941 vrcholů stanovena na **20% až 25%**. Při rozsáhlejších grafech může být optimální velikost i nižší. Pro experimenty byla stanovena 15% velikost (podle práce [2]).

- Optimální hodnota parametru p_f pro metodu Forest Fire byla stanovena na 0,7 až 0,9. Pro metodu Random Jump 0,2 a pro Divided Stratums 0,9. U metody Random Walk nad testovanými sítěmi na hodnotě parametru nezáleželo.

- Topologickou strukturu původní sítě nejlépe zachovává metoda **Divided Stratums**. Asortativitu a modularitu dobře zachovává i metoda **Random Walk**, ale její algoritmus nemusí navštívit odlehlejší části grafu.

Stávající implementace určitě poskytuje prostor pro zlepšení. V první řadě by mohl být použit jiný programovací jazyk, protože C# neposkytuje kvalitní open-source knihovny pro výpočty, které byly potřebné v implementaci. Lépe by si v této oblasti vedla implementace v C++, tím by se ušetřil čas. Budoucí rozšíření práce by mohlo zahrnovat vzorkování orientovaných grafů s ohodnocenými hranami. Rovněž by se budoucí práce na toto téma mohla zaměřit na odhadování vlastnosti původního grafu definováním přesných vztahů mezi konkrétní vlastností vzorku a původního grafu.

Literatura

[1] Most famous social network sites worldwide as of January 2017, ranked by number of active users (in millions). *Statista* [online]. 2017 [cit. 2017-04-12]. Dostupné z: `https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/`

[2] Sampling from large graphs. LESKOVEC, Jure a Christos FALOUTSOS. *KDD-2006: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* [online]. New York, NY: ACM Press, 2006, s. 631-636 [cit. 2017-01-29]. ISBN 1595933395. Dostupné z: `https://cs.stanford.edu/people/jure/pubs/sampling-kdd06.pdf`

[3] BARABÁSI, Albert-László. a Márton. PÓSFAI. *Network science*. Cambridge, United Kingdom: Cambridge University Press, 2016. ISBN 978-110-7076-266.

[4] ČERNÝ, Jakub. *Základní grafové algoritmy* [online]. KAM, MFF UK, 2010 [cit. 2017-01-09]. Dostupné z: `http://kam.mff.cuni.cz/~kuba/ka/ka.pdf`

[5] NEWMAN, Mark. *Networks: An Introduction*. New York: Oxford University Press, 2010. ISBN 01-992-0665-1.

[6] GLADWELL, Malcolm. *The tipping point: how little things can make a big difference*. Boston: Back Bay Books, 2002. ISBN 03-163-4662-4.

[7] Facts about Erdös Numbers and the Collaboration Graph. *The Erdös Number Project-Oakland University* [online]. [cit. 2017-01-19]. Dostupné z: `http://www.oakland.edu/enp/trivia/`

[8] BARABÁSI, Albert-László. *Linked: how everything is connected to everything else and what it means for business, science, and everyday life*. New York: Basic Books, 2014. ISBN 04-650-8573-3.

[9] Paretovo pravidlo (Pravidlo 80/20). *Management Mania* [online]. 2015 [cit. 2017-01-07]. Dostupné z: `https://managementmania.com/cs/paretovo-pravidlo`

[10] MILGRAM, Stanley. The Small-World Problem. *Psychology Today*. **1967**(1), 61-67. Dostupné také z: `http://barabasilab.neu.edu/courses/phys5116/content/milgram_smallworld.pdf`

[11] SLANINA, František a Miroslav KOTRLA. Sítě „malého světa“: Proč mají odlišné sítě podobnou strukturu? *Vesmír*. 2001, **2001**(11), 611-614. Dostupné také z: `http://casopis.vesmir.cz/files/file/fid/1198/aid/4791`

[12] BHAGAT, Smriti, Moira BURKE, Carlos DIUK, Ismail ONUR FILIZ a Sergey EDUNOV. Three and a half degrees of separation. In: *Facebook Research* [online]. 2016 [cit. 2017-01-13]. Dostupné z: https://research.fb.com/three-and-a-half-degrees-of-separation/

[13] GRANOVETTER, Mark S. The strength of weak ties. *American Journal of Sociology*. **1973**(78), 1360 - 1380. Dostupné také z: https://sociology.stanford.edu/sites/default/files/publications/the_strength_of_weak_ties_and_exch_w-gans.pdf

[14] Leonard Euler's Solution to the Konigsberg Bridge Problem. PAOLETTI, Teo. *Mathematical Association of America* [online]. [cit. 2017-04-24]. Dostupné z: http://www.maa.org/press/periodicals/convergence/leonard-eulers-solution-to-the-konigsberg-bridge-problem

[15] ERDŐS, Paul a Alfréd RÉNYI. On random graphs. I. *Publ. Math.* Debrecen, 1959, , 290-297.

[16] WATTS, Duncan J. a Steven H. STROGATZ. Collective dynamics of 'small-world' networks. *Nature*. 1998, **393**(6684), 440-442. Dostupné také z: http://www.nature.com/nature/journal/v393/n6684/full/393440a0.html

[17] KUDĚLKA, Miloš. *Metody analýzy dat* [online]. [cit. 2017-01-25]. Dostupné z: http://homel.vsb.cz/~kud007

[18] Data sampling. ROUSE, Margaret. *Search Business Analytics* [online]. [cit. 2017-04-22]. Dostupné z: http://searchbusinessanalytics.techtarget.com/definition/data-sampling

[19] CLAUSET, Aaron. *Network Analysis and Modeling, CSCI 5352: Sampled networks* [online]. In: . 2016 [cit. 2017-02-14]. Dostupné z: http://tuvalu.santafe.edu/~aaronc/courses/5352/csci5352_2016_L8b.pdf

[20] Total number of Websites & Size of the Internet as of 2013. *Facts Hunt* [online]. 2014 [cit. 2017-02-14]. Dostupné z: http://www.factshunt.com/2014/01/total-number-of-websites-size-of.html

[21] AL HASAN, Mohammad, Nesreen AHMED a Jennifer NEVILLE. *Network Sampling: Methods and Applications* [online]. In: . Chicago, 2013 [cit. 2017-02-14]. Dostupné z: https://www.cs.purdue.edu/homes/neville/courses/kdd13-tutorial.html

[22] MACHANAVAJJHALA, Ashwin a Jun YANG. Lab #11: Graph Sampling. In: *Everything Data* [online]. [cit. 2017-04-22]. Dostupné z: http://www.cs.duke.edu/courses/spring15/compsci216/lectures/lab11.pdf

[23] LESKOVEC, Jure, Jon KLEINBERG a Christos FALOUTSOS. Graph evolution. *ACM Transactions on Knowledge Discovery from Data* [online]. 2007, **1**(1), 2-es [cit. 2017-03-23]. DOI: 10.1145/1217299.1217301. ISSN 15564681. Dostupné z: `http://portal.acm.org/citation.cfm?doid=1217299.1217301`

[24] Expectation of Geometric Distribution. *Proof Wiki* [online]. [cit. 2017-04-24]. Dostupné z: `https://proofwiki.org/wiki/Expectation_of_Geometric_Distribution`

[25] GJOKA, Minas, Maciej KURANT, Carter T. BUTTS a Athina MARKOPOULOU. *A Walk in Facebook: Uniform Sampling of Users in Online Social Networks* [online]. 2009 [cit. 2017-02-21]. Dostupné z: `https://arxiv.org/pdf/0906.0060.pdf`

[26] WANG, Tianyi, Yang CHEN, Zengbin ZHANG, Tianyin XU, Long JIN, Pan HUI, Beixing DENG a Xing LI. Understanding Graph Sampling Algorithms for Social Network Analysis. In: *2011 31st International Conference on Distributed Computing Systems Workshops* [online]. IEEE, 2011, s. 123-128 [cit. 2017-02-21]. DOI: 10.1109/ICDCSW.2011.34. ISBN 978-1-4577-0384-3. Dostupné z: `http://ieeexplore.ieee.org/document/5961350/`

[27] DU, Xiaolin, Yunming YE, Yueping LI a Ge SONG. New Relational Networks Sampling Algorithm Using Topologically Divided Stratums. *Advanced Science and Technology Letters* [online]. 2014, **2014**(48), 108-119 [cit. 2017-02-25]. Dostupné z: `http://dx.doi.org/10.14257/astl.2014.48.19`

[28] RIBEIRO, Bruno a Don TOWSLEY. *Estimating and Sampling Graphs with Multidimensional Random Walks* [online]. 2010 [cit. 2017-03-05]. Dostupné z: `https://arxiv.org/pdf/1002.1751v2.pdf`

[29] HELD, Pascal, Benjamin KRAUSE a Rudolf KRUSE. *Dynamic Clustering in Social Networks using Louvain and Infomap Method* [online]. [cit. 2017-04-23]. Dostupné z: `https://arxiv.org/pdf/1603.02413.pdf`

[30] CHAKRABARTI, Deepayan a Christos FALOUTSOS. Graph mining: Laws, generators, and algorithms. *ACM Computing Surveys* [online]. 2006, **38**(1) [cit. 2017-01-30]. DOI: 10.1145/1132952.1132954. ISSN 03600300. Dostupné z: `http://www.cs.cmu.edu/~christos/PUBLICATIONS/acmCS06-graphs.pdf`

[31] FALOUTSOS, Michalis, Petros FALOUTSOS a Christos FALOUTSOS. On power-law relationships of the Internet topology. *ACM SIGCOMM Computer Communication Review* [online]. 1999, **29**(4), 251-262 [cit. 2017-01-30]. DOI: 10.1145/316194.316229. ISSN 01464833. Dostupné z: `http://www.cs.cmu.edu/~christos/PUBLICATIONS/sigcomm99.pdf`

[32] CHAPELA, Victor, Regino CRIADO, Santiago MORAL a Miguel ROMANCE. *Intentional risk management through complex networks analysis*. Springer, 2015. ISBN 978-3-319-26421-9.

[33] OCHODKOVÁ, Eliška. *Grafové algoritmy a komplexní sítě* [online]. [cit. 2017-04-23]. Dostupné z: `http://www.cs.vsb.cz/ochodkova/`

[34] DOROGOVTSEV, S. N., A. V. GOLTSEV, J. F. F. MENDES a A. N. SAMUKHIN. Spectra of complex networks. *Physical Review E* [online]. 2003, **68**(4), - [cit. 2017-04-02]. DOI: 10.1103/PhysRevE.68.046109. ISSN 1063-651x. Dostupné z: `http://link.aps.org/doi/10.1103/PhysRevE.68.046109`

[35] Social network analysis pro začátečníky. *Lupa.cz* [online]. [cit. 2017-04-18]. Dostupné z: `http://www.lupa.cz/clanky/social-network-analysis-pro-zacatecniky`

[36] Statistics - Kolmogorov Smirnov Test. In: *Tutorialspoint* [online]. [cit. 2017-03-04]. Dostupné z: `https://www.tutorialspoint.com/statistics/kolmogorov_smirnov_test.htm`

[37] *DBLP* [online]. [cit. 2017-03-12]. Dostupné z: `http://dblp.uni-trier.de/`

[38] SNAP: Network datasets. *High-energy physics citation network* [online]. [cit. 2017-03-12]. Dostupné z: `https://snap.stanford.edu/data/cit-HepPh.html`

[39] Datasets. *US Power Grid* [online]. [cit. 2017-03-12]. Dostupné z: `https://toreopsahl.com/datasets/#uspowergrid`

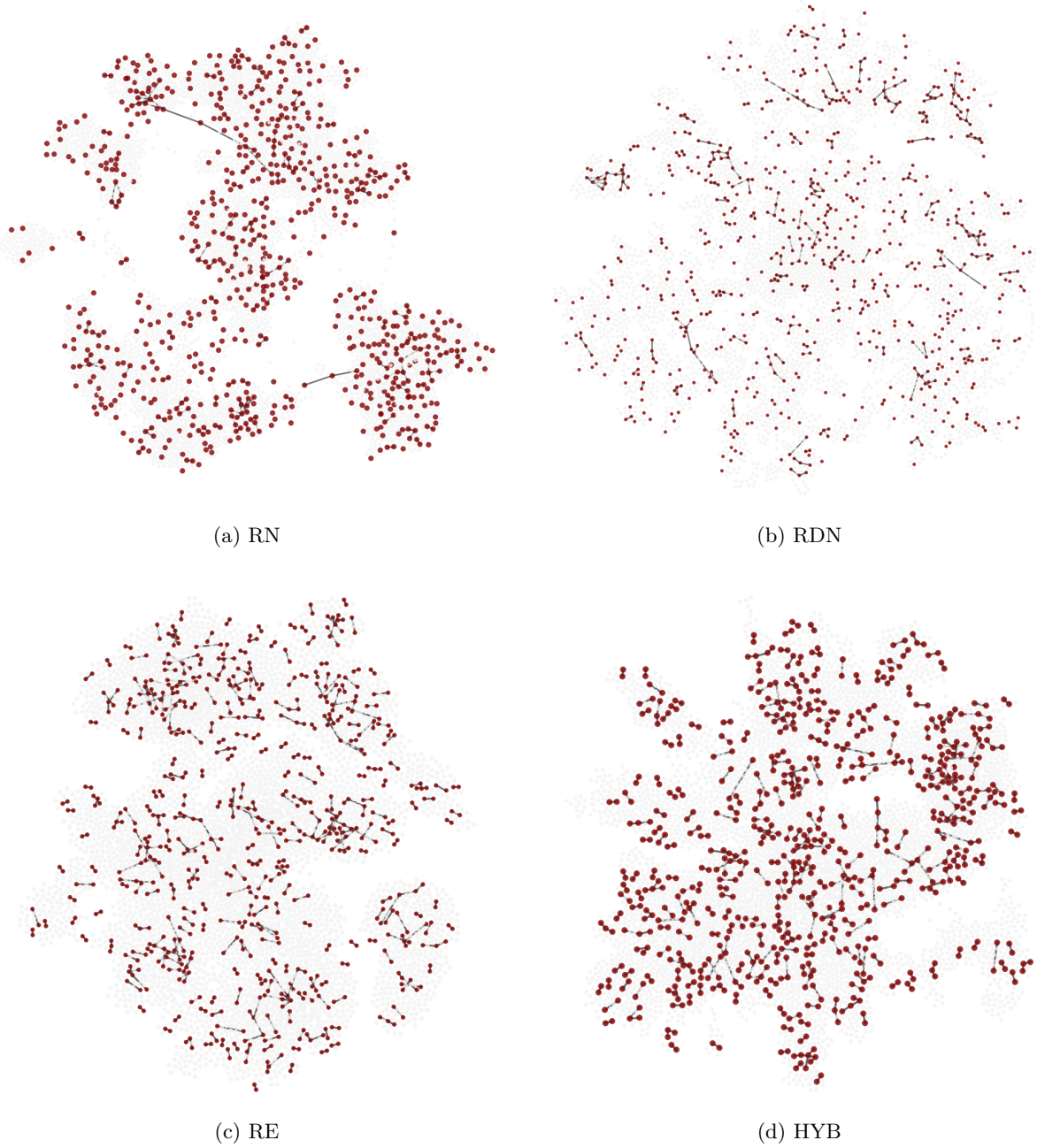
[40] KUDĚLKA, Miloš, Eliška OCHODKOVÁ a Šárka ZEHNALOVÁ. *Around Average Behavior: 3-lambda Network Model* [online]. In: . [cit. 2017-03-21]. Dostupné z: `https://arxiv.org/pdf/1701.01274.pdf`

[41] Enron email network. *SNAP* [online]. [cit. 2017-03-29]. Dostupné z: `https://snap.stanford.edu/data/email-Enron.html`

[42] NOLDUS, Rogier a Piet Van MIEGHEM. *Assortativity in Complex Network* [online]. 2015 [cit. 2017-04-05]. Dostupné z: https://www.nas.ewi.tudelft.nl/people/Piet/papers/JCN2015AssortativitySurveyRogier.pdf

[43] NEWMAN, M. E. J. Assortative Mixing in Networks. *Physical Review Letters* [online]. 2002, **89**(20), - [cit. 2017-04-02]. DOI: 10.1103/PhysRevLett.89.208701. ISSN 0031-9007. Dostupné z: `http://link.aps.org/doi/10.1103/PhysRevLett.89.208701`

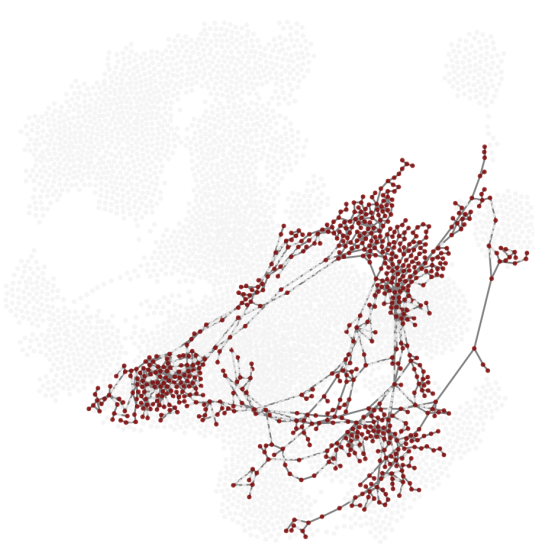
A Grafické znázornění vzorků



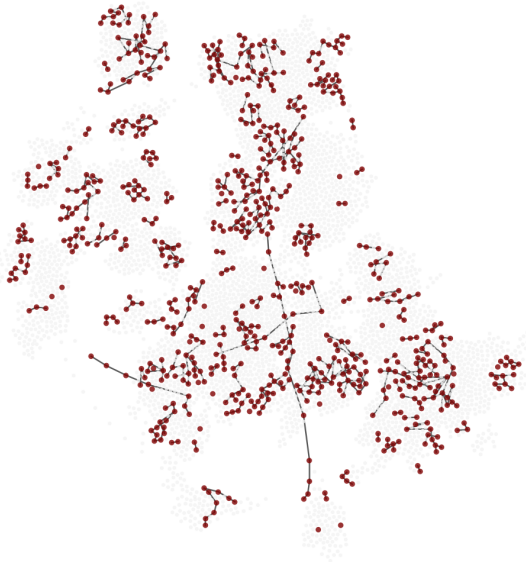
(a) RN (b) RDN

(c) RE (d) HYB

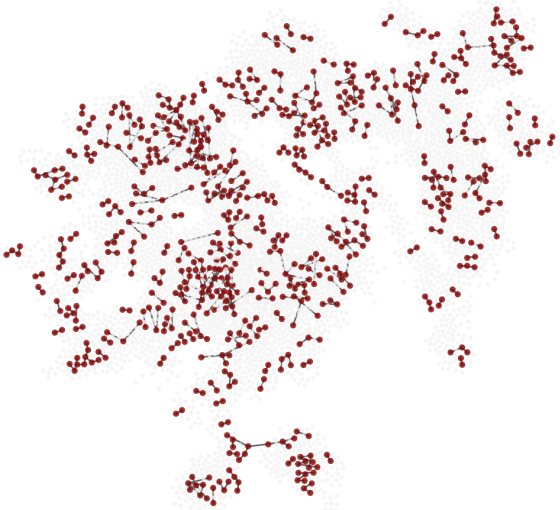
Obrázek 19: Vizualizace vzorků vytvořených vzorkovacími metodami založenými na náhodném výběru vrcholů a hran. Velikost vzorků je 15% původní velikosti **Enron sítě**.



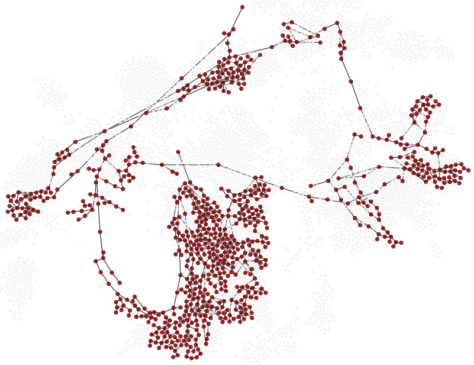
(a) RW



(b) RJ

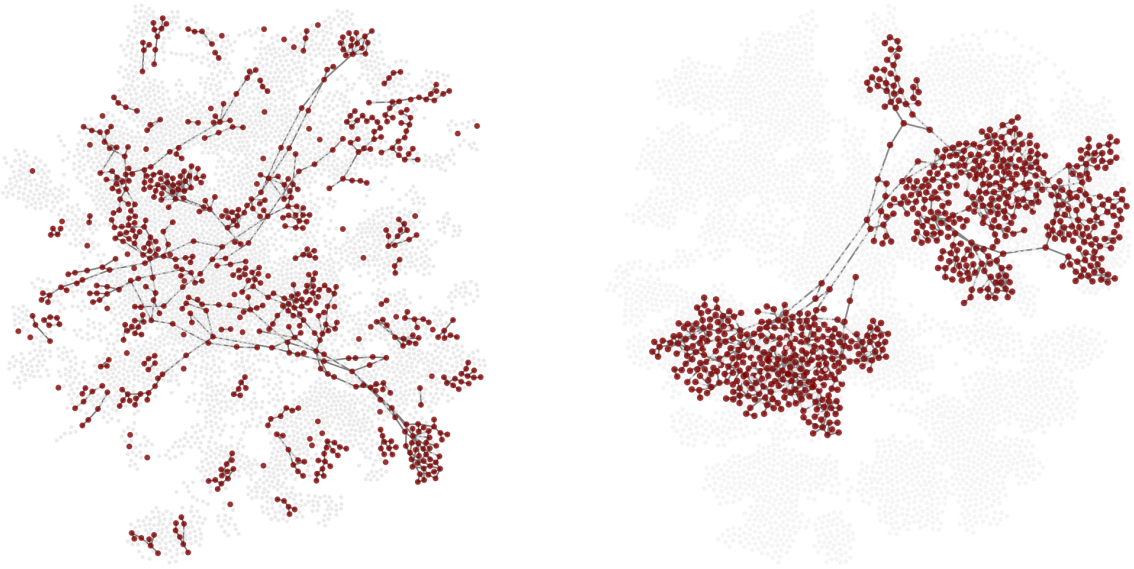


(c) FS



(d) MHRW

Obrázek 20: Vizualizace vzorků vytvořených vzorkovacími metodami Random Walk, Random Jump, Frontier Sampling a Metropolis Hastings Random Walk. Velikost vzorků je 15% původní velikosti **Enron sítě**.



(a) DS (b) FF

Obrázek 21: Vizualizace vzorků vytvořených vzorkovacími metodami Divided Stratums a Forest Fire. Velikost vzorků je 15% původní velikosti **Enron sítě**.

B Příloha na CD/DVD

- **dp.pdf** - text této práce
- **sources** - zdrojové kódy implementace
- **exe** - spustitelný program pro vytváření vzorků
- **experiments** - soubory vygenerované v experimentech